COMPARATIVE PERFORMANCE ANALYSIS OF LIGHTWEIGHT AND DEEP CNN MODELS FOR CONCRETE CRACK DETECTION

Badhon Singha

Abstract

Automated crack detection is essential for Structural Health Monitoring (SHM), yet most existing studies prioritise classification accuracy while overlooking computational efficiency critical for real-world deployment. This study presents a comprehensive comparative analysis of lightweight and deep convolutional neural network architectures for concrete crack detection, evaluating both predictive performance and computational tractability. Four pre-trained models MobileNetV2, VGG16, ResNet50, and InceptionV3 were evaluated under identical training conditions using a balanced dataset of 8,000 images. Results demonstrate that all architectures achieved high accuracy (0.98-1.00), with InceptionV3 attaining perfect classification. Remarkably, the lightweight MobileNetV2 achieved 0.99 accuracy, matching VGG16 and exceeding ResNet50, while exhibiting the lowest parameter count, fastest training time, and minimal inference cost. Confusion matrix analysis revealed that MobileNetV2 maintained a crack-class recall of 0.99 with minimal false negatives, ensuring safety-critical detection reliability. The findings demonstrate that architectural efficiency rather than network depth determines practical crack detection performance, establishing MobileNetV2 as optimal for resource-constrained, real-time structural health monitoring applications including edge devices, UAV platforms, and mobile inspection systems.

Received: *18 February, 2026*
Revised: *6 May, 2026*
Accepted: *20 July, 2026*
Published: *18 September, 2026*

Department of Civil Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh.

Corresponding email: *badhon10776@gmail.com*

DOI: *https://doi.org/10.54552/9zcj7631*

Keywords:
Concrete crack detection; Lightweight convolutional neural networks; Deep learning; Structural health monitoring; Computational efficiency

List of Notations:

AI	Artificial Intelligence	FN	False Negative	SHM	Structural Health Monitoring
CNN	Convolutional Neural Network	FP	False Positive	TN	True Negative
DCNN	Deep Convolutional Neural Network	GAP	Global Average Pooling	TP	True Positive
F1	F1-Score	GPU	Graphics Processing Unit	UAV	Unmanned Aerial Vehicle
FLOPs	Floating-Point Operations	ResNet	Residual Network	VGG	Visual Geometry Group

1.0 INTRODUCTION

Concrete is the most extensively used construction material in civil infrastructure owing to its strength, durability, and cost-effectiveness. Despite these advantages, concrete structures are inherently prone to deterioration caused by mechanical loading, environmental exposure, shrinkage, and chemical processes over time (Paul *et al.,* 2020) . Among the various forms of deterioration, surface cracking is one of the earliest and most visible indicators of structural distress. Cracks often signify the initiation of damage mechanisms such as reinforcement corrosion, fatigue, or material degradation, which, if left undetected, may progressively compromise structural safety and serviceability. Consequently, reliable crack detection plays a vital role in structural health monitoring (SHM) systems, enabling early intervention and preventive maintenance strategies (Farrar & Worden, 2007; Fu *et al.,* 2023).

Traditionally, crack detection has relied heavily on visual inspection conducted by trained engineers. While widely adopted in practice, manual inspection methods are inherently subjective and prone to human error, with outcomes strongly dependent on inspector experience and environmental conditions (Yeum & Dyke, 2015). Furthermore, such inspections are labor-intensive, time-consuming, and often impractical for large-scale infrastructure networks. Accessibility and safety challenges further limit their effectiveness, particularly for bridges, high-rise buildings, tunnels, and hazardous environments where direct inspection may pose significant risks. These limitations have motivated the development of automated, objective, and scalable crack detection techniques.

Advances in computer vision have significantly enhanced automated crack detection by enabling non-contact, image-based analysis of concrete surfaces with improved speed and objectivity compared to traditional manual inspection methods. Early computational approaches relied primarily on classical image processing techniques, including edge detection algorithms, intensity thresholding, and handcrafted feature extraction methods such as Haar wavelets, Gabor filters, and texture descriptors. While these conventional methods provided initial automation capabilities, they demonstrated limited robustness and generalisation when confronted with real-world challenges including varying illumination conditions, inconsistent surface textures, environmental noise, shadows, stains, and background clutter that visually resemble crack patterns (Yamaguchi & Hashimoto, 2006). The inherent dependence of these methods on manually designed features and fixed thresholding parameters constrained their adaptability across diverse concrete types and inspection environments, motivating the transition toward data-driven deep learning approaches capable of automatically learning hierarchical discriminative visual representations from labelled training data (Dorafshan *et al.,* 2018).

In contrast, deep learning particularly Convolutional Neural Networks (CNNs) has emerged as a powerful alternative due to its ability to automatically learn hierarchical and discriminative features directly from raw image data (LeCun *et al.,* 2015). CNN-based methods have consistently demonstrated superior performance over traditional image processing techniques in crack detection tasks, offering improved accuracy, generalisation capability, and resilience to environmental variations (Cha *et al.,* 2017; Yang *et al.,* 2018).

Despite the growing body of research on CNN-based crack detection, most existing studies prioritise classification accuracy as the dominant performance metric. However, accuracy alone does not fully reflect a model's suitability for real-world structural health monitoring (SHM) applications, where computational efficiency, processing latency, memory constraints, and power consumption are equally critical (Deng *et al.,* 2023). Most comparative studies fail to systematically examine the trade-off between predictive performance and computational cost, leaving practitioners without clear guidance on architecture selection for resource-constrained deployment scenarios such as edge computing devices, drone-mounted inspection systems, and mobile platforms (Cha *et al.,* 2017; Hsieh & Tsai, 2018). Additionally, limited attention has been given to lightweight architectures capable of enabling real-time processing and widespread deployment. This gap between laboratory performance and deployment feasibility motivates the need for comprehensive comparative analysis that evaluates both classification reliability and computational tractability across architectures of varying complexity. Computational efficiency, inference speed, and model complexity are critical factors for real-time and field-deployable systems, yet these aspects are often underreported or overlooked in the literature (Dorafshan *et al.,* 2018). Moreover, systematic comparisons between lightweight CNN architectures and deeper, computationally intensive models under identical experimental conditions remain limited. The implications of misclassification behaviour, particularly false negatives, which pose serious safety risks by allowing cracks to remain undetected, are also insufficiently addressed.

To address these gaps, this study presents a comprehensive comparative analysis of lightweight and deep CNN architectures for automated concrete crack detection. The investigation evaluates not only classification accuracy but also error behaviour and computational complexity to establish an accuracy–complexity trade-off framework. By integrating predictive performance with practical deployability considerations, this work aims to provide actionable guidance for selecting CNN models suitable for real-time, resource-constrained structural health monitoring applications.

1.1 CNN-Based Crack Detection

Concrete crack detection is conventionally modelled as a binary image classification task, where the objective is to discriminate between images containing structural defects (crack) and those representing intact or undamaged surfaces (no crack) (Li & Zhao, 2019).

In the evolving domain of structural health monitoring (SHM), defect identification has progressively shifted from labour-intensive manual inspections to automated, data-driven computer vision systems. Among various surface anomalies, concrete crack detection is most commonly formulated as an image-based classification problem, where deep learning models analyse visual information to determine the presence or absence of structural defects. This computational framing enhances inspection objectivity, efficiency, repeatability, and safety, while reducing human subjectivity inherent in traditional visual surveys.

1.1.1 Visual Characteristics and Challenges of Crack Detection

Cracks are among the earliest and most critical indicators of structural deterioration; however, their visual properties make automated detection particularly challenging. From an image analysis perspective, cracks exhibit several complex characteristics:

a) Thin and elongated morphology: Cracks generally appear as narrow, dark, and elongated lines or curves. In many cases, crack widths are extremely small (often below 0.1 mm), making them susceptible to loss during image resizing, compression, or downsampling.
b) Irregular and non-uniform geometry: Unlike engineered objects with predictable shapes, cracks possess irregular, discontinuous, and branching structures. Their length, width, orientation, and curvature vary stochastically, preventing the application of fixed geometric templates.
c) Low contrast and background interference: Cracks typically manifest as subtle texture discontinuities within complex backgrounds. Surface roughness, stains, shadows, moisture marks, joints, and debris often produce crack-like patterns, increasing the likelihood of false detections.

These factors collectively render crack detection a low-level visual recognition problem, where defect signals are weak and easily confused with background noise.

1.1.2 Automatic Spatial Feature Learning with CNNs

Conventional image processing techniques rely heavily on hand-crafted features, such as Sobel or Canny edge detectors, which require manual parameter tuning and exhibit high sensitivity to illumination changes, noise, and surface variability. Such limitations significantly restrict their robustness in real-world inspection scenarios.

Convolutional Neural Networks (CNNs) address these challenges by enabling automatic and adaptive feature learning directly from raw pixel data. Rather than relying on predefined rules, CNNs learn discriminative representations through hierarchical processing stages:

Convolutional layers apply learnable kernels to extract local spatial features. Shallow layers capture basic patterns such as edges, gradients, and textures, while deeper layers progressively encode more complex structures, including crack continuity and branching behavior.

Hierarchical representation learning allows CNNs to model both fine-grained local details and global contextual information. This multilevel abstraction provides robustness against variations in crack scale, orientation, and shape.

Weight sharing and local connectivity significantly reduce the number of trainable parameters while preserving spatial

correlations among neighboring pixels, enabling efficient learning even with limited datasets.

Through this layered architecture, CNNs develop a semantic understanding of crack morphology, making them particularly well-suited for complex surface defect detection tasks.

1.1.3 Binary Classification Formulation

In most practical implementations, CNN-based crack detection is framed as a binary classification problem. During training, labelled image datasets are divided into two categories:

a) Crack (Positive class): Images or image patches containing visible fractures.
b) No-crack (Negative class): Images representing intact or undamaged surfaces.

The features extracted by convolutional layers are passed to fully connected classification layers, where nonlinear activation functions, commonly Sigmoid or Softmax map the network output to a probability score between 0 and 1. A higher probability indicates a greater likelihood of crack presence, while lower values suggest a normal surface condition.

This binary formulation enables efficient screening of large volumes of structural imagery, allowing automated systems to rapidly filter non-defective regions and flag potential damage zones for further inspection, maintenance planning, or advanced segmentation analysis.

1.2 Deep CNN Architectures for Concrete Crack Detection

Modern computer vision for structural health monitoring has been redefined by Deep Convolutional Neural Networks (DCNNs), which automate the extraction of hierarchical features from raw pixel data. Unlike traditional image processing, DCNNs learn to recognise the intricate topologies and intensity variations of cracks through exposure to large, annotated datasets (Yang & Huang, 2024).

1.2.1 VGG-Style Deep Networks

The Visual Geometry Group (VGG) architecture serves as a foundational benchmark for crack classification problems. VGG models, such as VGG-16 and VGG-19, are characterised by their uniform architecture and simplicity, making them easy to implement and understand.

The Visual Geometry Group (VGG) architecture, particularly the VGG-16 and VGG-19 iterations, remains a seminal benchmark in the domain of computer vision for structural health monitoring. Its utility in crack classification is rooted in a highly uniform architectural design that prioritises depth over breadth. By exclusively utilising deep stacks of 3×3 convolutional filters with a fixed stride of 1 and consistent padding ("same" padding), VGG networks systematically increase the effective receptive field without sacrificing spatial resolution prematurely (Karen Simonyan & Zisserman, 2014).

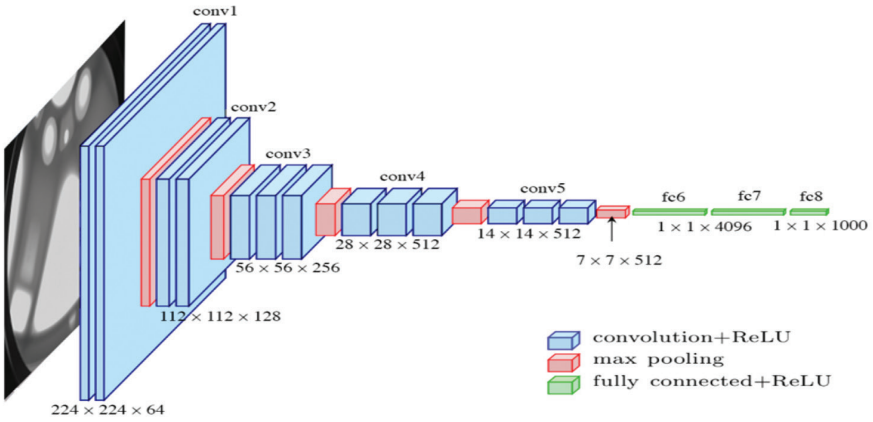

Figure 1: The standard VGG-16 network architecture (Ferguson et al., 2017)

Theoretically, stacking two 3×3 convolutional layers yields an effective receptive field of 5×5, while three layers approximate a 7×7 field; however, the former approach introduces additional non-linearities (rectified linear units) and reduces the total number of parameters, enhancing the network's ability to learn discriminative features. In the context of crack detection, this hierarchical structure is critical, as the initial layers capture low-level primitives such as edges, gradients, and corners, while deeper strata synthesise these primitives to identify complex crack textures and fracture patterns indicative of structural failure (Cha *et al.,* 2017).

Despite its robustness as a baseline for feature extraction, the VGG architecture is plagued by significant inefficiencies regarding computational economy. The primary drawback stems from its massive parameter density, largely driven by the three fully connected layers that follow the convolutional base. For instance, VGG-19 contains approximately 143 million parameters, a scale that necessitates substantial memory bandwidth and computational power during both training and inference (Karen Simonyan & Zisserman, 2014). This computational intensity limits the feasibility of deploying VGG models in resource-constrained environments or on edge devices required for real-time structural assessment. When compared to modern architectures such as ResNet or EfficientNet, which utilise residual connections or compound scaling to achieve higher accuracy with a fraction of the parameters, VGG is often viewed as computationally prohibitive for large-scale or latency-sensitive industrial applications (Tan & Le, 2020). Consequently, while VGG serves as an excellent pedagogical tool and a stable benchmark for texture-based classification tasks like crack detection, its practical utility is often overshadowed by the prohibitive cost of its deep, dense connectivity (Zhang *et al.,* 2024).

1.3 Lightweight CNN Architectures for Structural Inspection

The transition of structural health monitoring (SHM) from centralized workstations to autonomous, in-situ applications has necessitated the development of Convolutional Neural Network (CNN) architectures that prioritise computational efficiency without compromising diagnostic fidelity. This section explores the theoretical motivations and architectural mechanisms of lightweight models tailored for real-time crack detection.

1.3.1 Motivation for Lightweight Models

Traditional manual crack detection methodologies are increasingly regarded as non-scalable, labour-intensive, and inherently subjective, leading to inconsistent safety evaluations and significant temporal delays in infrastructure assessment (Altaf *et al.,* 2025). To facilitate immediate decision-making in maintenance workflows and enable proactive intervention strategies, there is an imperative need for real-time inspection systems capable of providing instant feedback during on-site structural examinations (Tang *et al.,* 2019). This requirement is particularly acute for field deployment on edge computing platforms, such as smartphones, embedded systems, and microcontrollers, where devices must operate independently of high-latency cloud infrastructure to ensure continuous functionality in remote or connectivity-limited environments (Su *et al.,* 2024).

The integration of Unmanned Aerial Vehicle (UAV)-based edge computing represents a transformative approach in modern structural health monitoring, enabling rapid inspection of large-scale infrastructure and access to structurally challenging locations. However, UAV deployment is severely constrained by limited onboard battery capacities, restricted payload weight allowances, and the critical need to maximise operational flight time for comprehensive coverage. These operational constraints necessitate computationally efficient deep learning models with minimal parameter counts, reduced inference latency, and low power consumption characteristics that enable extended autonomous operation without sacrificing detection reliability (Deng *et al.,* 2023). Lightweight CNN architectures specifically designed for resource-constrained environments thus emerge as essential enablers of practical, scalable, and economically viable automated crack detection systems suitable for widespread deployment across civil infrastructure networks (Hsieh & Tsai, 2018).

The use of UAV-based edge computing is highly relevant in modern structural health monitoring, yet it is severely constrained by limited battery capacities and the need to maximise operational flight time (Roy *et al.,* 2025). Standard deep CNN architectures, while highly accurate in laboratory settings, are often impractical for field use because they are "resource-hungry," placing excessive demands on memory bandwidth and Graphics Processing Unit (GPU) cycles (Altaf *et al.,* 2025). Consequently, the development of compact models is driven by the strategic necessity to balance predictive accuracy with energy-efficient deployability on resource-constrained embedded hardware.

1.3.2 MobileNet-Based Lightweight Networks

The core philosophy of lightweight design, exemplified by the MobileNet architecture family, centres on the optimization of the accuracy-to-latency trade-off for mobile and embedded vision tasks (Chen *et al.,* 2025).This optimisation is primarily achieved through the depthwise separable convolution concept, which mathematically factors a standard convolution into two distinct operations: a depthwise convolution for spatial feature extraction per channel and a pointwise convolution for channel-wise integration (Raza *et al.,* 2025).

This structural factorisation enables a significant parameter and computation reduction, typically yielding an 8- to 9-fold decrease in Floating-Point Operations (FLOPs) compared to standard convolutions (Altaf *et al.,* 2025). Advanced iterations, such as MobileNetV2, further enhance efficiency by introducing inverted residual linear bottlenecks, which expand feature representations in a high-dimensional space before projecting them back to a low-dimensional bottleneck to preserve semantic signals.

These architectural refinements make MobileNet variants exceptionally suitable for binary crack detection, as they deliver robust sensitivity to structural defects while maintaining a minimal memory footprint (Wang *et al.,* 2025). Such models are capable of achieving high-speed inference reporting latencies as low as 11 to 12 milliseconds per image on edge devices like the Raspberry Pi 4 or NVIDIA Jetson Nano (Altaf *et al.,* 2025; Raza *et al.,* 2025).

1.3.3 Feature Learning Capability of Lightweight Models

Despite their reduced depth and parameter count, lightweight models demonstrate a sophisticated ability to capture localised crack features, such as fine edges, contours, and texture discontinuities (Karimi *et al.,* 2024). By utilising hierarchical spatial representations, these models effectively identify the high-frequency patterns associated with micro-cracks that are often lost during the downsampling processes of heavier networks (Nasimov & Cho, 2025). Literature indicates that compact architectures can achieve comparable performance to deep CNNs, with specialised lightweight configurations reaching crack-class F1-scores of approximately 0.73 to 0.90+ on benchmark datasets (Altaf *et al.,* 2025; Raza *et al.,* 2025).

Furthermore, lightweight designs exhibit a reduced risk of overfitting on surface textures by utilising mechanisms such as Global Average Pooling (GAP) and systematic regularisation (Wang *et al.,* 2025). By replacing dense, parameter-heavy fully connected layers with GAP, these models minimise redundant parameters, forcing the network to focus on discriminative crack morphologies rather than memorising noisy background textures or irrelevant surface roughness. This makes compact CNNs highly generalisable across diverse materials such as concrete, asphalt, and masonry while maintaining the operational efficiency required for proactive maintenance in smart city infrastructures (Raza *et al.,* 2025; Wang *et al.,* 2025).

In the contemporary academic landscape of Structural Health Monitoring (SHM), the analysis of existing crack detection literature reveals a systemic preference for high-capacity architectures over deployment-ready models. While research has successfully demonstrated the efficacy of artificial intelligence in infrastructure assessment, several critical gaps remain regarding the balance between precision and operational efficiency.

1.3.4 Hegemony of Deep Convolutional Neural Networks (DCNNs)

Deep Convolutional Neural Networks (DCNNs) currently maintain a dominant position in crack detection research, frequently serving as the benchmark for automated feature extraction (Krishnan *et al.,* 2025). High-capacity architectures such as VGG-16, ResNet-50, and Inception V3 are standard choices due to their ability to learn complex hierarchical textures. For instance, recent studies utilising Inception V3 have achieved classification accuracies

as high as 99.98% in binary crack detection tasks. This dominance is largely attributed to the robust discriminative power of these deep structures, which can isolate subtle structural anomalies that elude traditional image processing techniques(Qian, 2023).

1.3.5 The Computational Gap: Lightweight and Efficient Models

Despite the rise of mobile robotics and Unmanned Aerial Vehicles (UAVs) in inspections, lightweight models designed from first principles remain significantly less explored than their larger counterparts (Altaf *et al.,* 2025).

While specialised architectures like MobileNetV2 and EfficientNet-Lite are gaining traction, many models utilised in the literature remain computationally demanding and unsuitable for edge deployment. (Altaf *et al.,* 2025) observe that achieving real-time performance on resource-constrained embedded systems requires a shift toward computational parsimony, yet such architectures are often viewed as secondary to "research-grade" deep models. Furthermore, the lack of exploration into bespoke lightweight networks tailored specifically for multi-class substrate classification represents a persistent lacuna in the field.

1.3.6 Metric Imbalance and the Neglect of Efficiency

A pervasive trend in existing studies is an overwhelming emphasis on classification accuracy, precision, and F1-scores, often to the exclusion of real-world deployment parameters (Egodawela *et al.,* 2024). Traditional methodologies frequently prioritise high accuracy metrics in controlled environments while rarely discussing the associated computational footprint.

Research conducted by Egodawela *et al.,* (2024) notes that while increasing the depth of a convolutional network enhances accuracy, these models are "extremely taxing on hardware" and often fail to meet the latency requirements for real-time UAV-assisted inspections. Consequently, there is a critical need for a multidimensional evaluation strategy that integrates inference speed, memory footprint, and recall-specific metrics alongside general accuracy.

1.3.7 Lack of Standardised Benchmarking and Fair Comparison

Finally, a significant hurdle to objective evaluation is the lack of fair comparisons conducted under identical conditions. The academic community currently lacks a universally accepted framework for benchmarking, as researchers frequently utilise disparate datasets, varied image resolutions, and idiosyncratic hardware setups. This lack of standardisation results in inconsistent performance claims that are difficult to replicate or compare across different studies.

Moreover, the variability in lighting conditions, surface roughness, and camera angles is seldom standardised, leading to potential biases in model performance reports. Establishing shared benchmark datasets with unified performance metrics is essential for bridging the gap between theoretical research and industrial implementation.

2.0 METHODOLOGY

The study uses a data-driven experimental framework to objectively compare the performance of lightweight and deep neural network architectures for concrete crack detection. A balanced image dataset consisting of 8,000 concrete surface images was used, including 4,000 cracked and 4,000 non-crack samples. All images were resized to 224×224 pixels to ensure compatibility across CNN architectures and to maintain consistent input dimensionality. Pixel values were normalised to the [0,1] range to improve numerical stability during optimisation.

To enable unbiased performance assessment, the dataset was partitioned into training (70%), validation (15%), and testing (15%) subsets while preserving class balance across all splits. This fixed partitioning strategy was applied uniformly to all models to ensure a fair and reproducible comparison.

2.1 CNN Architecture Selection and Model Configuration

Four pre-trained convolutional neural network architectures representing distinct design paradigms were selected to systematically investigate the relationship between architectural complexity and crack detection performance, as illustrated in Figure 2. The selected models span from lightweight to deep networks: MobileNetV2 employs depth-wise separable convolutions for computational efficiency; VGG16 utilises uniform convolutional blocks with progressive depth; ResNet50 incorporates residual connections to improve gradient propagation in deep structures; and InceptionV3 applies parallel multi-scale convolutional pathways to capture features at different spatial resolutions. This selection enables comprehensive evaluation of how architectural depth, parameter magnitude, feature extraction strategy, and computational design influence crack detection capability.

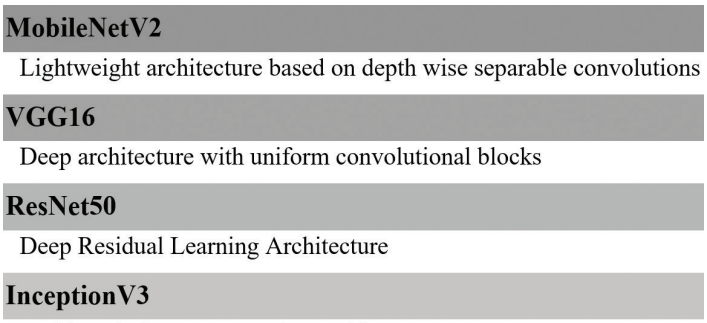

Figure 2: Pre-trained Convolutional Neural Network Models

All networks were initialised with ImageNet pre-trained weights to leverage transferable low-level and mid-level visual representations learned from large-scale image classification tasks. A unified binary classification head, consisting of global average pooling followed by fully connected layers with sigmoid activation, was appended to each architecture to ensure structural consistency. The convolutional backbone layers were frozen during training, allowing only the classification head parameters to be optimised. This controlled configuration preserves pre-trained feature extraction while ensuring that observed performance differences arise from inherent architectural characteristics rather than variations in fine-tuning strategy, thereby enabling objective comparison of CNN paradigms for concrete crack identification.

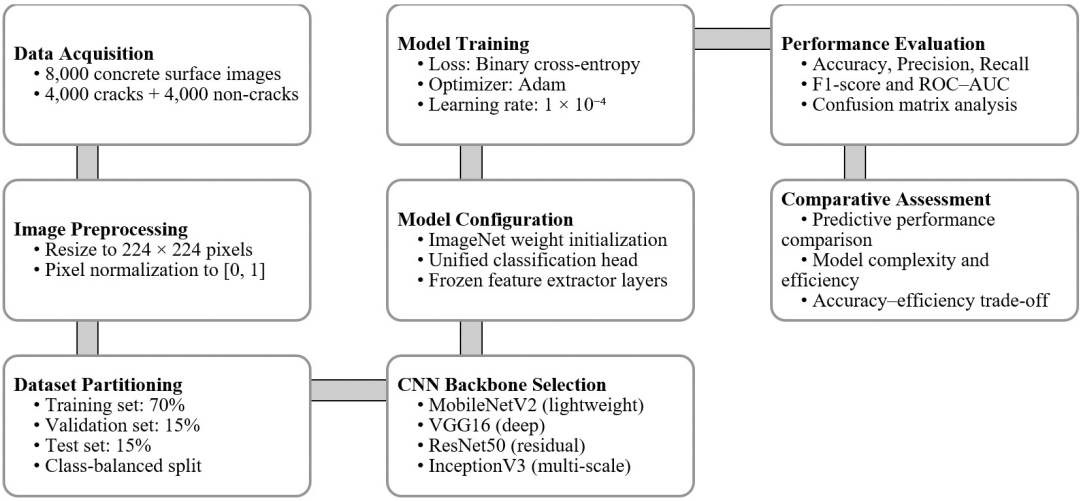

Figure 3: Experimental workflow for comparative evaluation of CNN-based concrete crack detection models

2.2 Model Training Strategy

Model training was conducted under strictly controlled and identical experimental conditions across all evaluated architectures to ensure fair and unbiased performance comparison. A binary cross-entropy loss function was employed as the optimisation objective, appropriate for the binary classification task distinguishing between cracked and non-cracked concrete surfaces. The Adam optimiser was selected for gradient-based parameter updates due to its adaptive learning rate properties and robust convergence characteristics across diverse network architectures. A fixed learning rate of 1×10^{-4} was applied uniformly to all models, selected to provide stable convergence across architectures of varying depth and complexity.

All models were trained using identical batch sizes and for the same number of epochs to maintain uniform exposure to training data. Validation set performance was continuously monitored after each epoch to track learning progression and ensure consistent learning behaviour across different architectures. This standardised training protocol eliminates confounding effects arising from hyper-parameter variation, enabling direct and objective comparison of architectural capabilities in learning discriminative features for concrete crack detection. By isolating architectural differences as the sole variable, the experimental design ensures that performance variations can be reliably attributed to fundamental differences in feature extraction mechanisms rather than artifacts of the training procedure.

2.3 Performance Evaluation and Comparative Analysis

A comprehensive multi-criteria evaluation framework was established to systematically assess and compare the performance of the selected CNN architectures across both predictive accuracy and computational efficiency dimensions. Model performance was quantitatively evaluated on the independent test set, which comprised 15% of the total dataset (800 images with balanced class distribution). Standard classification metrics were computed from confusion matrix analysis, including overall accuracy, precision, recall, and F1-score. These metrics provide complementary perspectives on model performance: accuracy measures overall correctness as (TP + TN) / (TP + TN + FP + FN), precision quantifies the reliability of positive predictions as TP / (TP + FP), recall assesses the completeness of crack detection as TP / (TP + FN), and F1-score represents the harmonic mean balancing precision and recall as 2 × (precision × recall) / (precision + recall), where TP, TN, FP, and FN denote True Positive(TP), True Negative(TN), false positives, and false negatives, respectively.

Beyond aggregate metrics, misclassification patterns were examined in detail through analysis of false positive and false negative distributions to assess crack detection reliability and identify potential failure modes. False positive errors, where non-cracked concrete surfaces are incorrectly classified as cracked, were analysed to assess the risk of unnecessary inspections, while false negative errors, representing actual cracks misclassified as non-cracked surfaces, were scrutinised with particular attention given their critical safety implications in structural health monitoring.

In addition to predictive performance, computational efficiency metrics were systematically quantified to characterise the accuracy-complexity trade-off, including total parameter count (representing model size and memory requirements), training time per epoch (reflecting computational burden during model development), and inference cost (indicating latency and resources required for real-time prediction). The complete experimental workflow from data acquisition (8,000 images: 4,000 cracked and 4,000 non-cracked), through image preprocessing (224 × 224 pixel resizing and normalization), dataset partitioning (70% training, 15% validation, 15% test), model configuration (ImageNet initialisation, frozen backbone, unified classification head), CNN backbone selection (MobileNetV2, VGG16, ResNet50, InceptionV3), model training (binary cross-entropy loss, Adam optimizer, learning rate 1×10^{-4}), to comparative performance assessment is systematically illustrated in Figure 3. This integrated framework enables identification of optimal models that balance superior crack detection reliability with computational efficiency suitable for practical deployment in real-world structural health monitoring applications.

3.0 RESULT AND ANALYSIS

This section presents the quantitative performance of the selected CNN architectures on the independent test dataset. The comparative evaluation focuses on classification accuracy, precision, recall, and F1-score to assess crack detection reliability across lightweight and deep models.

3.1 Classification Performance on Test Dataset

The primary objective of this study was to assess and compare the predictive effectiveness of lightweight and Deep Convolutional Neural Network (CNN) architectures for automated concrete crack detection. All models were trained under a standardised learning framework using the Adam optimizer and binary cross-entropy loss to ensure a fair and unbiased comparison. Performance evaluation was conducted on an independent test set comprising 15% of the total dataset, corresponding to 800 images equally distributed between cracked and non-cracked classes.

The quantitative results indicate that all four selected architectures MobileNetV2, VGG16, ResNet50, and InceptionV3 exhibit excellent performance in extracting discriminative features and accurately classifying concrete surface conditions. The overall classification accuracies achieved by each model are summarised in Table 1.

Among the evaluated models, InceptionV3 achieved the highest classification accuracy, attaining a perfect score of 1.00 on the test set. This superior performance can be attributed to its multi-scale feature extraction mechanism, which enables effective capture of crack geometries with varying widths, orientations, and surface textures.

Table 1: Accuracy Comparison of Evaluated CNN Models

Model	Architecture Paradigm	Test Accuracy
InceptionV3	Deep (Multi-scale Feature Extraction)	1.00
MobileNetV2	Lightweight (Depth-wise Separable Convolution)	0.99
VGG16	Deep (Uniform Convolutional Blocks)	0.99
ResNet50	Deep (Residual Learning)	0.98

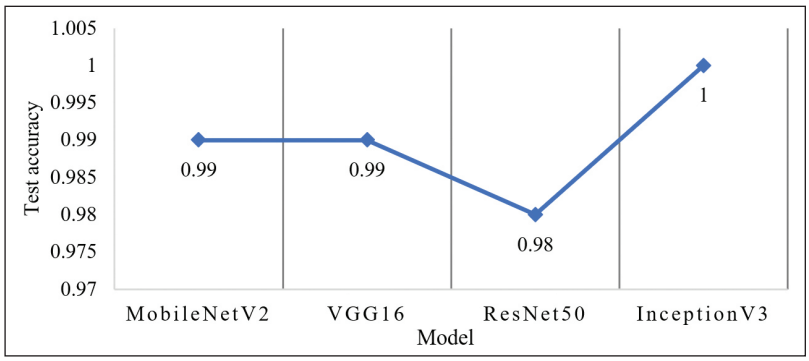

Figure 4: Model accuracy comparison

Table 2: Class-Wise Performance Metrics for Crack Detection

Model	Class	Precision	Recall	F1-Score	Support
InceptionV3	Non-Crack (0)	1.00	0.99	1.00	400
	Crack (1)	1.00	1.00	1.00	400
MobileNetV2	Non-Crack (0)	0.99	0.99	0.99	400
	Crack (1)	0.99	0.99	0.99	400
VGG16	Non-Crack	0.99	0.99	0.99	400
	Crack	0.99	0.99	0.99	400
ResNet50	Non-Crack (0)	0.99	0.98	0.98	400
	Crack (1)	0.98	0.99	0.99	400

A notable outcome of this analysis is the marginal performance difference between the lightweight MobileNetV2 and the deeper, more computationally demanding architectures. Despite its reduced parameter count and reliance on depth-wise separable convolutions, MobileNetV2 achieved an accuracy of 0.99, matching the performance of VGG16 and outperforming ResNet50, which recorded the lowest accuracy of 0.98. This finding highlights that high classification reliability can be achieved without excessive architectural depth. The classification performance across all evaluated CNN models is highly consistent, with a maximum accuracy variation of only 2%. While InceptionV3 delivers the highest predictive accuracy, MobileNetV2 demonstrates a superior accuracy-to-complexity balance, making it particularly suitable for real-time crack detection applications in resource-constrained and field-deployable environments.

3.2 Class-Wise Performance Evaluation

While overall accuracy provides a general overview of model performance, a class-wise evaluation is essential to understand how each architecture handles the specific features of cracked versus non-cracked surfaces. This analysis utilises Precision, Recall, and F1-Score derived from the confusion matrix of the 800-image test set, which contains a balanced distribution of 400 cracked and 400 non-cracked samples.The detailed class-wise metrics for MobileNetV2, InceptionV3, ResNet50, and VGG16 are presented in Table 2.

3.2.1 Analysis of Metrics and Detection Difficulty

The class-wise evaluation metrics indicate that all evaluated CNN architectures demonstrate strong and consistent classification performance across both crack and non-crack classes. Among them, InceptionV3 achieved near-perfect results, maintaining a score of 1.00 for precision, recall, and F1-score across both classes.

This highlights its superior capability in learning robust and discriminative features from concrete surface imagery. Both MobileNetV2 and VGG16 exhibited identical and highly balanced performance, achieving precision and recall values of 0.99 for both crack and non-crack classes. Such consistency confirms their reliability in distinguishing complex surface textures and subtle crack patterns, despite architectural differences in depth and computational complexity. Detecting the crack class is generally more challenging due to variations in crack width, irregular orientations, inconsistent lighting conditions, and the presence of surface debris that can visually resemble cracks.

In this regard, ResNet50 displayed the greatest variability among the evaluated models. While it achieved a high recall of 0.99 for the crack class, indicating effective crack identification, it recorded a slightly lower precision of 0.98. This suggests a higher tendency toward false positives, where non-cracked surfaces were incorrectly classified as cracked, compared to the other architectures.

3.2.2 Confusion Matrix and Error Analysis

To move beyond aggregate accuracy and gain deeper insight into model reliability, a detailed analysis of misclassification behaviour was conducted using confusion matrix results. This evaluation focuses on False Positive (FP) and False Negative (FN) errors derived from the balanced test dataset comprising 800 images (400 crack and 400 non-crack samples).

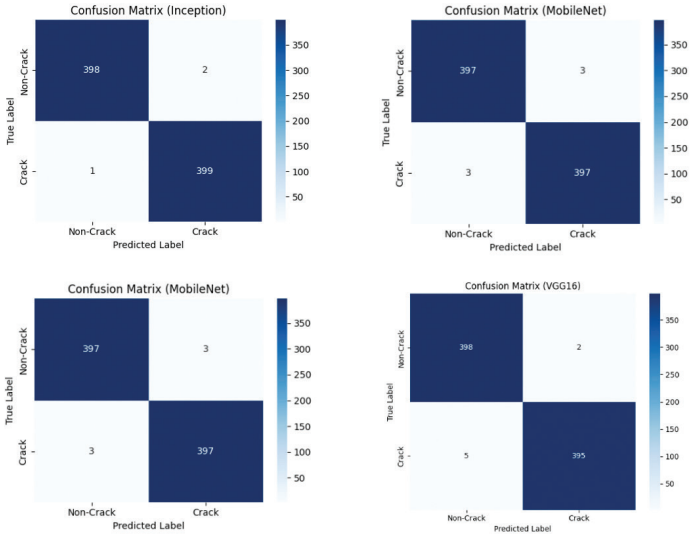

Figure 5: Confusion matrix based error analysis of the evaluated CNN models

3.2.3 False Positive (FP) Trends

A false positive occurs when a non-cracked concrete surface is incorrectly classified as cracked. While such errors do not directly compromise structural safety, they may lead to unnecessary inspections and increased maintenance costs. Among the evaluated architectures, ResNet50 exhibited the highest tendency toward false positives. This behaviour is reflected in its slightly lower recall for the non-crack class (0.98) and reduced precision for the crack class (0.98), indicating that surface irregularities or texture noise were occasionally misinterpreted as cracks.

In contrast, InceptionV3 demonstrated the lowest false positive rate, achieving a perfect crack-class precision of 1.00. This result highlights the effectiveness of its multi-scale feature extraction mechanism in discriminating true cracks from visually similar non-crack patterns.

3.2.4 False Negative (FN) Trends and Missed Crack Analysis

False negatives represent the most critical error type in structural health monitoring, as they correspond to actual cracks being misclassified as non-cracked surfaces. The confusion matrix analysis reveals that:

InceptionV3 achieved a recall of 1.00 for the crack class, successfully identifying all 400 cracked samples without any missed detections. MobileNetV2, VGG16, and ResNet50 each recorded a crack-class recall of 0.99, indicating a very low false negative rate of approximately 1%.

Although the performance gap among the models is marginal, the identical low false negative rates observed for MobileNetV2 and VGG16 are particularly noteworthy. This suggests that the depth-wise separable convolution strategy employed by MobileNetV2 is highly effective in preserving crack-related features, despite its significantly reduced architectural complexity.

3.2.5 Engineering Implications and Risk Assessment

From a structural engineering standpoint, the consequences of false negatives are substantially more severe than those of false positives. While a false positive may only result in redundant manual inspection, a missed crack can allow ongoing structural fatigue or deterioration to remain undetected. Over time, such unnoticed defects may propagate and lead to serious serviceability issues or, in extreme cases, catastrophic structural failure if timely maintenance is not performed.

Accordingly, the zero false-negative performance of InceptionV3 positions it as the most robust option for safety-critical assessment scenarios. However, the high crack-detection recall achieved by MobileNetV2 (0.99), combined with its low computational demand, indicates that it offers a practical and reliable balance between safety assurance and efficiency. This makes it particularly suitable for real-time and resource-constrained structural monitoring applications.

3.3 Lightweight vs. Deep CNN Performance Comparison

A key objective of this study is to examine the balance between architectural complexity and predictive reliability by directly comparing the lightweight MobileNetV2 with deeper CNN paradigms, namely VGG16, ResNet50, and InceptionV3. Despite substantial differences in depth, parameter count, and feature extraction strategy, all models exhibited consistently high performance on the balanced test dataset of 800 images.

3.4 Performance Comparison across Architectures

The experimental results reveal that increased architectural depth does not result in a proportional improvement in classification accuracy for concrete crack detection. InceptionV3, which employs multi-scale feature extraction, achieved the highest test accuracy of 1.00, indicating its strong capability in capturing crack patterns of varying sizes and orientations.

Remarkably, the lightweight MobileNetV2 achieved an accuracy of 0.99, which is identical to that of the significantly deeper VGG16 and higher than that of ResNet50, which recorded the lowest accuracy at 0.98. This close performance clustering demonstrates that high detection accuracy can be attained without reliance on deep or computationally expensive network designs.

3.4.1 Feature Extraction Efficiency of Lightweight Architecture

The strong performance of MobileNetV2 is particularly notable given its reliance on depth-wise separable convolutions, a design choice intended to minimise computational cost while preserving representational power. The model achieved precision and recall values of 0.99 for both crack and non-crack classes, equaling or surpassing the performance of deeper architectures. This indicates that MobileNetV2 efficiently captures the localised and high-contrast geometric features characteristic of surface cracks, despite having substantially fewer parameters.

In contrast, deeper architectures such as ResNet50 and VGG16 did not consistently translate their increased depth into superior detection capability, suggesting diminishing returns for additional layers when applied to surface-level defect identification tasks. A critical insight from this comparative analysis is that increasing model depth does not inherently guarantee improved crack detection performance.

While InceptionV3 offers a marginal accuracy advantage, the fact that MobileNetV2 outperformed the deeper ResNet50 (0.99 vs. 0.98 accuracy) highlights the effectiveness of optimised lightweight architectures. For applications involving localised geometric patterns such as concrete cracks, architectural efficiency appears to be more influential than sheer network depth.

3.4.2 Computational Efficiency and Model Complexity

While classification accuracy is essential for crack detection reliability, the practical applicability and deployability of CNN-based inspection systems depend critically on computational efficiency characteristics. Real-world structural health monitoring applications, particularly those involving edge computing devices, mobile platforms, or resource-constrained field environments, require models that balance high detection accuracy with computational tractability.

To comprehensively evaluate the accuracy-complexity trade-off inherent in different architectural designs, three key computational indicators were systematically quantified: (1) total parameter count, representing model size and memory footprint; (2) training time per epoch, reflecting the computational burden during model development and iterative refinement; and (3) inference cost, indicating the latency and processing requirements for real-time prediction on new images.

These metrics collectively determine whether a given architecture is suitable for deployment in time-sensitive, resource-limited, or large-scale monitoring scenarios. The comparative results are illustrated in Figure 6, Figure 7 while a detailed quantitative comparison is presented in Table 3.

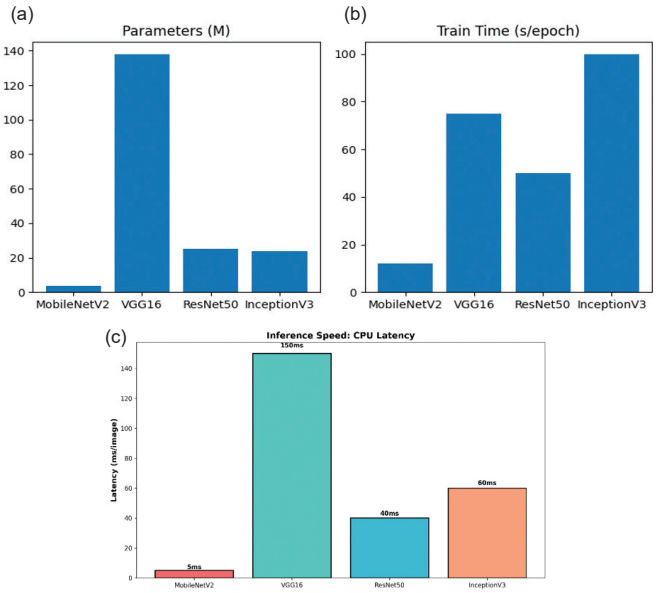

Figure 6: Computational efficiency comparison of CNN architectures showing (a) model parameter counts and (b) training time per epoch (c) inference speed

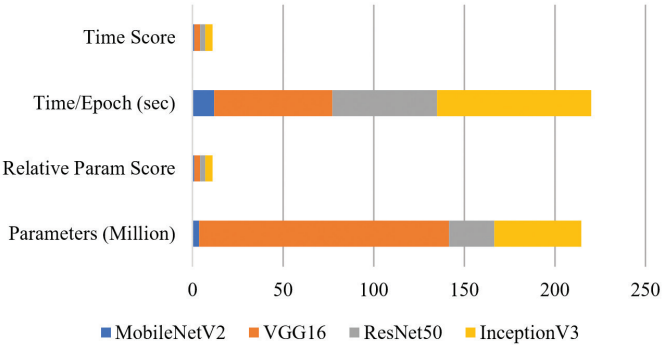

Figure 7: Comparison of Model Parameters Across CNN Architectures

3.4.3 Computational Complexity Comparison

In addition to predictive accuracy, the operational feasibility of CNN-based crack detection systems depends on computational efficiency. For structural health monitoring applications such as drone-assisted inspections, embedded edge devices, and large-scale infrastructure surveillance models must deliver reliable predictions while maintaining low memory usage and minimal processing latency. To quantitatively assess the trade-off between performance and complexity, three computational indicators were evaluated: total parameter count (model size and memory demand), training time per epoch (computational load during model development), and inference cost (real-time prediction latency).

Table 3: Accuracy Complexity Trade-Off Analysis of CNN Models

Model	Architecture Type	Parameters (Million)	Training Time / Epoch (s)	Inference Cost (GigaFLOPs / Image)	Test Accuracy
MobileNetV2	Lightweight	3.5	10–15	0.001	0.99
VGG16	Deep	138	60–90	15	0.99
ResNet50	Deep	25.6	45–60	4	0.98
InceptionV3	Deep (Multi-scale)	23.8	75–120	6	1.00

The comparative results, summarised in Figure 6 and Table 3, reveal significant variation in architectural efficiency. MobileNetV2 demonstrates the smallest parameter count and fastest processing speed due to its depth-wise separable convolution mechanism. Despite its compact structure, it achieved a test accuracy of 0.99, equivalent to VGG16 and exceeding ResNet50. This finding indicates that efficient feature extraction can be achieved without extensive network depth or excessive parameterisation.

Conversely, VGG16 and ResNet50 exhibit substantially higher parameter counts and longer training and inference times. While VGG16 attained comparable accuracy (0.99), its computational demand is considerably greater. ResNet50, although designed to enhance deep network training through residual connections, recorded the lowest accuracy (0.98) among the evaluated models, demonstrating that increased architectural complexity does not necessarily yield proportional gains in crack detection performance.InceptionV3's multi-scale feature extraction technique resulted in the greatest classification accuracy (1.00).

However, this marginal improvement over MobileNetV2 is associated by the biggest parameter count, longest training time, and highest inference cost. Such additional processing burden may introduce latency limits, limiting its utility in resource-constrained or real-time monitoring environments.

In general, the analysis demonstrates that lightweight systems can deliver near-optimal detection performance while retaining significant computational savings. For actual structural health monitoring systems needing scalability and low-latency operation, models with decreased complexity offer a more balanced and deployable solution than deeper, resource-intensive networks.

4.0 CONCLUSION

This study conducted a comparative evaluation of four pre-trained convolutional neural network architectures MobileNetV2, VGG16, ResNet50, and InceptionV3 for automated concrete crack detection using a balanced dataset of 8,000 images. All models demonstrated high classification capability, achieving accuracies between 0.98 and 1.00. Among them, InceptionV3 achieved the highest accuracy (1.00), while MobileNetV2 and VGG16 achieved 0.99, and ResNet50 achieved 0.98.

Despite large differences in architectural complexity, the performance variation among the models was minimal. Notably, MobileNetV2 achieved nearly the same predictive performance as the deeper architectures while maintaining the lowest parameter count, fastest training time, and minimal inference cost. The model also maintained a high crack-class recall of 0.99, indicating reliable crack detection capability.

In the end, the findings show that lightweight CNN designs can offer noticeably higher computational efficiency and detection performance that is similar to deeper networks. These results demonstrate that MobileNetV2 is a very useful model for real-time and resource-constrained structural health monitoring applications, whereas deeper architectures like InceptionV3 might be better when the highest level of classification accuracy is needed.

RESEARCH FUNDING

The author received no financial support for this research or the publication of this article. ■

REFERENCES

[1] Altaf, A., Mehmood, A., Filograno, M. L., Alharbi, S., & Iqbal, J. (2025). Deployable deep learning models for crack detection: Efficiency, interpretability, and severity estimation. *Buildings, 15*(18), 3362. https://doi.org/10.3390/buildings15183362

[2] Cha, Y., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering, 32*(5), 361–378. https://doi.org/10.1111/mice.12263

[3] Chen, G., Xia, Z., Ma, X., Jiang, Y., & He, Z. (2025). MobileNet-GDR: A lightweight algorithm for grape leaf disease identification based on improved MobileNetV4-small. *Frontiers in Plant Science*. https://doi.org/10.3389/fpls.2025.1702071

[4] Deng, Y., Ju, H., Zhong, G., & Li, A. (2023). Data quality evaluation for bridge structural health monitoring based on deep learning and frequency-domain information. *Structural Health Monitoring, 22*(5), 2925–2947. https://doi.org/10.1177/14759217221138724

[5] Dorafshan, S., Thomas, R. J., & Maguire, M. (2018). SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks. *Data in Brief, 21*, 1664–1668. https://doi.org/10.1016/j.dib.2018.11.015

[6] Egodawela, S., Khodadadian Gostar, A., Buddika, H. A. D. S., Dammika, A. J., Harischandra, N., Navaratnam, S., & Mahmoodian, M. (2024). A deep learning approach for surface crack classification and segmentation in unmanned aerial vehicle assisted infrastructure inspections. *Sensors, 24*(6), 1936. https://doi.org/10.3390/s24061936

[7] Farrar, C. R., & Worden, K. (2007). An introduction to structural health monitoring. Philosophical Transactions of the Royal Society A: Mathematical, *Physical and Engineering Sciences, 365*(1851), 303–315. https://doi.org/10.1098/rsta.2006.1928

[8] Ferguson, M., Ak, R., Lee, Y.-T. T., & Law, K. H. (2017). Automatic localization of casting defects with convolutional neural networks. In *2017 IEEE International Conference on Big Data (Big Data)* (pp. 1726–1735). IEEE. https://doi.org/10.1109/BigData.2017.8258115

[9] Fu, R., Cao, M., Novák, D., Qian, X., & Alkayem, N. F. (2023). Extended efficient convolutional neural network for concrete crack detection with illustrated merits. *Automation in Construction, 156*, 105098. https://doi.org/10.1016/j.autcon.2023.105098

[10] Hsieh, Y.-A., & Tsai, Y. J. (2020). Machine learning for crack detection: Review and model performance comparison. *Journal of Computing in Civil Engineering, 34*(5), 04020038. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000918

[11] Karimi, N., Mishra, M., & Lourenco, P. (2024). Automated surface crack detection in historical constructions with various materials using deep learning-based YOLO network. *International Journal of Architectural Heritage, 19*, 1–17. https://doi.org/10.1080/15583058.2024.2376177

[12] Krishnan, S. S. R., Karuppan, M. K. N., Khadidos, A. O., Khadidos, A. O., Selvarajan, S., Tandon, S., & Balusamy, B. (2025). Comparative analysis of deep learning models for crack detection in buildings. *Scientific Reports, 15*(1), 2125. https://doi.org/10.1038/s41598-025-85983-3

[13] Li, S., & Zhao, X. (2019). Image-based concrete crack detection using convolutional neural network and exhaustive search technique. *Advances in Civil Engineering, 2019*, 6520620. https://doi.org/10.1155/2019/6520620

[14] Nasimov, R., & Cho, Y. I. (2025). Smart city infrastructure monitoring with a hybrid vision transformer for micro-crack detection. *Sensors, 25*(16), 5079. https://doi.org/10.3390/s25165079

[15] Paul, S., Van Zijl, G., & Šavija, B. (2020). Effect of fibers on durability of concrete: A practical review. *Materials, 13*(20), 4562. https://doi.org/10.3390/ma13204562

[16] Qian, Y. (2023). Performance comparison among VGG16, InceptionV3, and ResNet on galaxy morphology classification. *Journal of Physics: Conference Series, 2580,* 012009. https://doi.org/10.1088/1742-6596/2580/1/012009

[17] Raza, A., Hanif, F., & Mohammed, H. A. (2025). Efficient crack and surface-type recognition via CNN-block development mechanism and edge profiling. *Scientific Reports, 15*(1), 40073. https://doi.org/10.1038/s41598-025-25956-8

[18] Roy, S., Yogi, B., Majumdar, R., Ghosh, P., & Das, S. K. (2025). Deep learning-based crack detection and prediction for structural health monitoring. *Discover Applied Sciences, 7*(7), 674. https://doi.org/10.1007/s42452-025-07272-y

[19] Su, G., Qin, Y., Xu, H., & Liang, J. (2024). Automatic real-time crack detection using lightweight deep learning models. Engineering *Applications of Artificial Intelligence, 138*, 109340. https://doi.org/10.1016/j.engappai.2024.109340

[20] Tan, M., & Le, Q. V. (2020). EfficientNet: Rethinking model scaling for convolutional neural networks. *arXiv*. https://doi.org/10.48550/arXiv.1905.11946

[21] Tang, Z., Chen, Z., Bao, Y., & Li, H. (2019). Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. *Structural Control and Health Monitoring, 26*(1), e2296. https://doi.org/10.1002/stc.2296

[22] Wang, X., Zhang, F., & Zou, X. (2025). Efficient lightweight CNN and 2D visualization for concrete crack detection in bridges. *Buildings, 15*(18), 3423. https://doi.org/10.3390/buildings15183423

[23] Yamaguchi, T., & Hashimoto, S. (2006). Automated crack detection for concrete surface image using percolation model and edge information. In *IECON 2006: 32nd Annual Conference on IEEE Industrial Electronics* (pp. 3355–3360). IEEE. https://doi.org/10.1109/IECON.2006.348070

[24] Yang, L., & Huang, E. (2024). Structural health monitoring data analysis using deep learning techniques. In *Proceedings of the 2024 8th International Conference on Electronic Information Technology and Computer Engineering* (pp. 913–927). Association for Computing Machinery. https://doi.org/10.1145/3711129.3711286

[25] Yeum, C. M., & Dyke, S. J. (2015). Vision-based automated crack detection for bridge inspection. *Computer-Aided Civil and Infrastructure Engineering, 30*(10), 759–770. https://doi.org/10.1111/mice.12141

[26] Zhang, J., Xia, H., Li, P., Zhang, K., Hong, W., & Guo, R. (2024). A pavement crack detection method via deep learning and a binocular-vision-based unmanned aerial vehicle. *Applied Sciences, 14*(5), 1778. https://doi.org/10.3390/app14051778

PROFILE



BADHON SINGHA is a Civil Engineering graduate from Ahsanullah University of Science and Technology (AUST), Bangladesh. His academic and research interests include seismic performance of structures, base isolation systems, and modular construction in disaster-prone regions. He has authored and co-authored multiple research papers, including studies on fluid viscous dampers, lead rubber bearing isolators, and soft-story structural behaviour, several of which are accepted or under review in peer-reviewed journals.
Email address: badhon10776@gmail.com