Journal of Advanced Intelligent Computing and Informatics
(JAICI)

Volume 1 Issue 1
Article Number: JAICI20260004

An Explainable Transfer Learning Framework with Adaptive Image Augmentation for Concrete Surface Crack Classification

Vinay Deeti* [1] and Shilpa Bade-Gite[2]

[1]Independent Researcher, United States
[2]University of York Mumbai, Mumbai, Maharashtra, India

September 8, 2026

Abstract

Concrete surface crack classification is an important component of condition assessment, yet image-based models often remain vulnerable to dataset homogeneity, environmental variation, opaque decision logic, and deployment constraints. This paper develops the Explainable Adaptive Transfer Learning Concrete Crack Classification Framework (EATL-Crack), a conceptual framework synthesized from peer-reviewed open-access literature rather than from newly conducted experiments. The framework links source-aware dataset preparation, image-condition assessment, condition-responsive augmentation, ImageNet-based transfer learning, staged fine-tuning, calibrated binary classification, visual explanation, corruption testing, cross-dataset evaluation, and efficiency-aware deployment selection. Its augmentation controller selects transformation type, magnitude, and probability from observed limitations in illumination, blur, noise, orientation, perspective, texture, and crack scale while restricting operations that could erase or physically distort fine cracks. The learning module compares conventional and lightweight convolutional backbones and separates head training from controlled upper-layer adaptation. Grad-CAM or Grad-CAM++ is used to audit whether predictions rely on crack paths rather than shadows, joints, stains, borders, or aggregate texture, while confidence calibration supports referral of uncertain images to human inspectors. Literature synthesis indicates that transfer learning and realistic augmentation can reduce overfitting and accelerate adaptation, but also shows that high within-dataset accuracy does not establish cross-environment reliability. EATL-Crack therefore provides a transparent research and implementation blueprint whose operational value requires prospective empirical validation.

Keywords: Adaptive Image Augmentation; Concrete Crack Classification; Explainable Artificial Intelligence; Infrastructure Inspection; Robustness Evaluation; Transfer Learning

1. Introduction

Concrete is central to transportation systems, buildings, bridges, tunnels, dams, and other public assets. Surface cracking is not a complete diagnosis of structural condition, but it is an accessible sign of shrinkage, corrosion-related expansion, restraint, overloading, environmental exposure, or progressive deterioration. Timely recognition can prioritise closer inspection and maintenance. Reviews of vision-based crack detection emphasise both its broad infrastructure relevance and the difficulty of obtaining reliable decisions from uncontrolled images [1], [2].

*Corresponding Author: Vinay Deeti (deeti.vinay@gmail.com)
Received: 01 July 2026; Revised: 31 July 2026; Accepted: 04 September 2026; Published: 30 September 2026
© 2026 The Author(s).
This is an open access article licensed under a Creative Commons Attribution-NonCommercial 4.0 International Licence (CC BY-NC 4.0).

Conventional visual inspection depends on trained personnel and remains essential for engineering judgement, but repeated surveys are slow, costly, subjective, and inconsistently documented. Bridge soffits, façades, tunnel crowns, dams, and traffic-exposed locations can also require closures, elevated access, or unmanned platforms. Traditional image processing reduces some effort, yet edge and threshold methods remain sensitive to parameter choice, noise, illumination, and rough backgrounds [3].

CNNs shifted crack analysis from handcrafted features toward learned representations. Early classification work demonstrated crack/non-crack transfer learning [4], and fully convolutional methods extended learning to localisation and segmentation [5]. Later studies reused pretrained networks, fine-tuned task-specific layers, and combined augmentation with concrete imagery [6], [7]. However, limited or homogeneous datasets allow models to exploit texture, borders, acquisition patterns, or background cues; feature-silencing research confirms that irrelevant content can influence crack predictions [8].

Transfer learning is attractive because concrete datasets are much smaller than generic image corpora. It offers useful initial representations, reduces random-initialisation burden, and often accelerates convergence. Comparative studies report strong results from pretrained backbones and fine-tuning [9–12], but random splits can be optimistic when neighbouring patches or repeated surfaces cross partitions. Classification, localisation, segmentation, and geometric measurement are distinct tasks [12], [14], and SDNET2018 illustrates the importance of transparent source composition across decks, walls, and pavements [16].

Augmentation increases apparent diversity and can simulate deployment variation, but uniform policies are poorly matched to crack imagery. Cropping may remove a thin crack, interpolation may blur its edge, warping may alter width, and unsuitable flips may create implausible patterns. Conversely, insufficient brightness, blur, compression, or perspective variation leaves models unprepared for field cameras. Explainability is also necessary to determine whether predictions rely on cracks rather than shadows, joints, stains, aggregate boundaries, or borders.

The research gap is therefore not the absence of another backbone comparison, but the absence of an integrated and auditable framework that connects condition-aware augmentation, staged transfer learning, calibrated confidence, explanation inspection, robustness testing, cross-dataset evaluation, and deployment efficiency. This paper addresses that gap by proposing EATL-Crack as a literature-supported blueprint for binary concrete surface crack classification. It does not claim experimental validation. Its contributions are:

1) a modular framework that connects data quality, learning, assurance, and deployment decisions for concrete surface crack classification;

2) an adaptive augmentation strategy in which image conditions and dataset deficiencies govern transformation choice, magnitude, and probability;

3) a staged transfer-learning procedure that separates classification-head learning from controlled fine-tuning of upper backbone layers;

4) an explainability module based principally on Grad-CAM or Grad-CAM++ for spatial auditing of model attention;

5) a robustness, confidence-calibration, and cross-dataset evaluation strategy designed to expose domain shift and unsafe uncertainty; and

6) a deployment-oriented selection procedure that balances crack recall and complementary predictive measures against latency, memory, parameter count, model size, and explanation quality.

2. Related Literature

2.1. Traditional Concrete Crack Detection

Traditional pipelines separate crack pixels from background using global or adaptive thresholding, Sobel or Canny edges, morphology, skeletonisation, and connected-component rules. Texture descriptors, local binary patterns, Gabor filters, wavelets, and handcrafted shape features can support classification or measurement. These methods are interpretable and efficient on uniform surfaces, but their thresholds fail under shadows, stains, joints, pores, exposed aggregate, moisture, uneven illumination, and camera noise [1–3]. Multiresolution analysis improves scale handling but still requires careful filtering and post-processing [12].

2.2. Deep Learning for Concrete Surface Inspection

Deep models learn hierarchical visual features directly from labelled images. Patch classifiers indicate crack presence, detectors identify regions, and segmentation networks predict pixels. Classification is cheaper to annotate; localisation

and segmentation support mapping and measurement at higher labelling cost. CNNs combine edges, texture, and context and commonly outperform handcrafted pipelines [4], [5], [10], [14]. Their weakness is overfitting: high-capacity models can exploit correlations that remain stable within one split but fail on another structure or camera [8], [13].

2.3. Transfer Learning

Transfer learning reuses representations learned from large source datasets. A frozen backbone is economical but may preserve mismatch between natural-image features and fine crack patterns. Partial fine-tuning adapts selected upper blocks and often provides a practical balance. Complete fine-tuning can increase adaptation but needs more data and may overwrite useful pretrained features. Concrete comparisons of VGG, residual, dense, mobile, inception, and efficient backbones show that ranking depends on dataset, training protocol, and resource objective rather than size alone [6], [9], [11], [14].

VGG provides a conventional, easily understood baseline but has many parameters. ResNet introduces identity shortcuts that stabilise deep optimisation [17]; DenseNet promotes feature reuse through dense connectivity [18]. VGG-style depth remains influential in transfer studies [19], while Inception combines multi-scale processing paths [20]. MobileNetV2 uses inverted residuals and linear bottlenecks for constrained devices [21]. EfficientNet jointly scales depth, width, and resolution to improve efficiency [22], and ConvNeXt modernises convolutional design while retaining a pure CNN structure [23]. These generic architecture papers do not establish superiority for concrete cracks; rather, they define candidates whose suitability must be tested under source-aware splits, calibration, robustness, and device constraints.

2.4. Image Augmentation

Conventional augmentation includes flips, small rotations, translations, scaling, cropping, brightness and contrast changes, blur, noise, and perspective transformation. Random erasing discourages dependence on narrow regions [24]; AutoAugment learns policy combinations [25]; MixUp interpolates images and labels [26]; and CutMix replaces a region while adjusting the target [27]. These methods can regularise learning, but crack semantics constrain them: an operation should not erase the only crack, broaden a hairline defect, or create implausible fragments. Policy choice must therefore reflect image quality, crack scale, acquisition conditions, and missing dataset variation.

2.5. Explainable Artificial Intelligence

Gradient-weighted class activation mapping (Grad-CAM) projects class-discriminative information from a convolutional layer back to image space [28]. Grad-CAM++ refines gradient weighting and can improve localisation when multiple relevant regions are present [29]. Integrated gradients attributes a prediction by accumulating gradients along a baseline-to-input path [30], while SHAP connects feature attribution to Shapley-value principles [31]. Saliency approaches are useful for model auditing, but their visual plausibility is not sufficient evidence of faithfulness; sanity checks have shown that some explanation maps can remain visually similar after model or label randomisation [32]. For crack inspection, explanations should therefore be treated as diagnostic evidence, compared across methods where feasible, and assessed against crack masks or expert-reviewed regions rather than presented as proof of correctness.

2.6. Identified Research Gap

The reviewed literature contains strong individual components: pretrained crack classifiers, segmentation networks, data augmentation, uncertainty modelling, and weakly supervised localisation. What remains fragmented is the assurance chain linking them. Many studies use fixed augmentation, random within-dataset splits, uncalibrated probabilities, limited corruption testing, and accuracy-centred model selection. Few frameworks jointly ask whether augmentation is physically credible, whether attention falls on crack paths, whether confidence is calibrated, whether performance transfers across sources, and whether the selected model fits a target edge platform. EATL-Crack is designed to make those questions explicit and sequential.

3. Research Methodology

This paper uses a structured, integrative literature synthesis covering concrete crack classification and related localisation, segmentation, uncertainty, transfer learning, augmentation, explainability, robustness, and inspection studies. It prioritises peer-reviewed open-access work from 2018 onward and adds seminal architecture and method papers.

Preprint-only sources, general websites, commercial material, and dataset pages without peer-reviewed papers were excluded.

The synthesis compared data source and split design, transfer strategy, augmentation, evaluation measures, generalisation evidence, and implementation constraints. Because studies differ in resolution, class balance, task definition, and partitioning, their numerical results were not pooled. Recurring practices and limitations were instead organised into a workflow linking source-aware separation, realistic augmentation, controlled fine-tuning, calibration, visual auditing, robustness, and efficiency. The method is integrative rather than a claimed exhaustive systematic review.

4. Proposed EATL-Crack Framework

EATL-Crack denotes the Explainable Adaptive Transfer Learning Concrete Crack Classification Framework. Its workflow is: concrete surface image acquisition → data-quality assessment → preprocessing → adaptive augmentation selection → transfer-learning backbone selection → staged training and fine-tuning → crack classification → confidence calibration → explainability analysis → robustness and cross-dataset evaluation → deployment recommendation. The framework is modular so that each decision can be documented, audited, and replaced without obscuring the remaining pipeline. The complete modular workflow and the interaction between the data-adaptation, learning, assurance, and deployment stages are illustrated in Fig. 1.

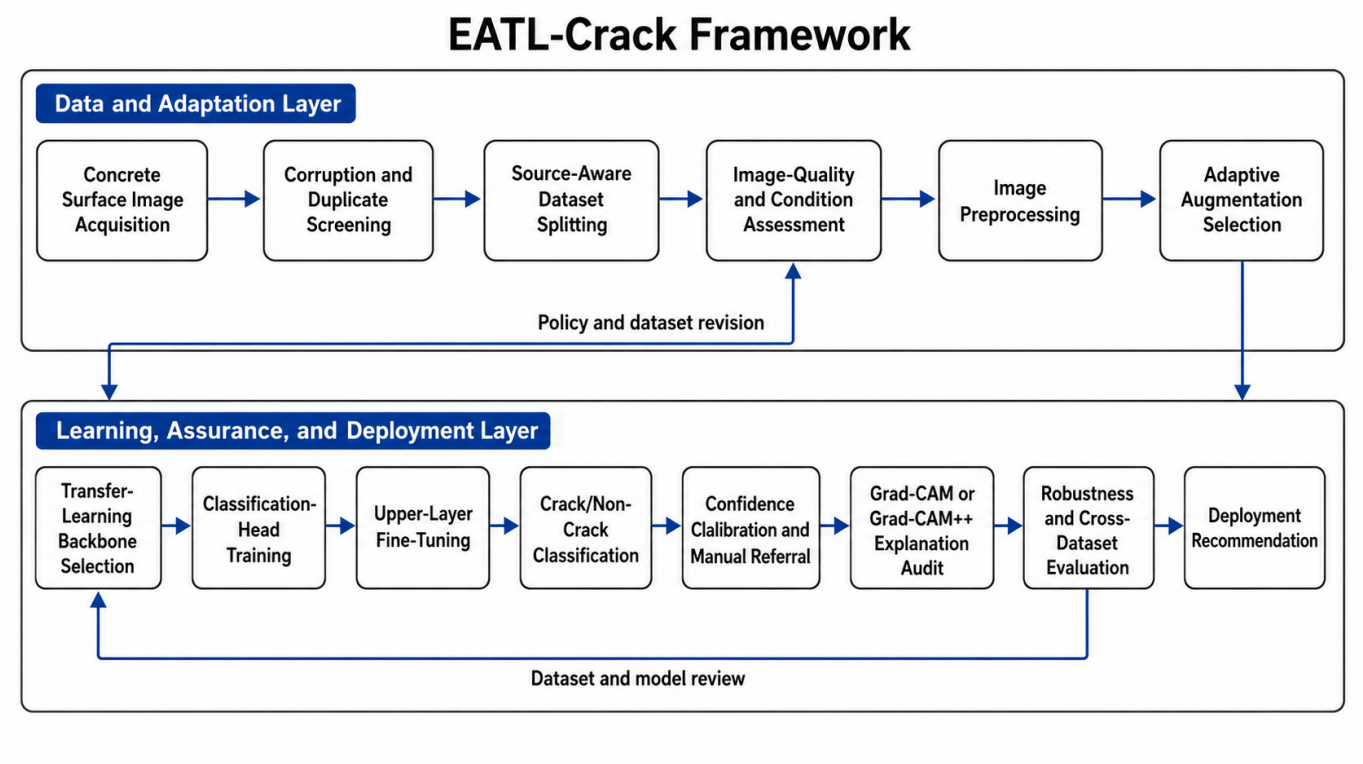


Figure 1: Conceptual architecture of the EATL-Crack framework.

4.1. Image Acquisition and Dataset Preparation

Images may come from peer-reviewed public datasets, handheld or fixed cameras, smartphones, inspection vehicles, or unmanned aerial vehicles. Corrupted files, exact and near duplicates must be removed before augmentation. Splits should occur before transformation and, where crops share a photograph or structure, at source-image or structure level. Class balance must be recorded and handled transparently through sampling, weighting, or loss design.

A dataset manifest should retain source, structure identifier, camera, resolution, scale, angle, and environmental conditions where available. It should also record illumination, texture, blur, noise, shadow, moisture, stains, crack width, and orientation. These fields enable subgroup analysis and distinguish transfer from repeated acquisition conditions. An untouched test set and an external source should be retained whenever possible.

4.2. Image-Quality and Condition Assessment

A preliminary module estimates augmentation-relevant conditions. Luminance statistics describe brightness and contrast; focus measures estimate blur; local residuals approximate noise; and metadata records resolution or scale. Crack orientation and thickness may come from labels, weak masks, line filters, or a lightweight estimator. Texture, shadow, stain, and perspective indicators can use image statistics or a small learned model. The aim is to characterise missing variation, not to make the final crack decision.

4.3. Adaptive Image-Augmentation Module

The augmentation controller maps condition indicators and dataset deficiencies to a bounded policy. Narrow illumination motivates moderate brightness and contrast changes; mobile or aerial capture motivates realistic blur, noise, compression, and resolution reduction; limited orientation diversity motivates small rotations; and oblique imaging motivates perspective transforms. MixUp or CutMix may be tested at low probability because a crack can occupy a tiny region. Large rotations, strong warping, heavy erasing, and aggressive cropping are restricted when they remove, widen, or fragment cracks.

For an input image x, an augmented sample is represented as

$$x' = T_k(x; \theta_k) \tag{1}$$

where T_k is a selected transformation and θ_k is its magnitude. The controller samples k, θ_k, and application probability from condition vector q(x) and dataset summary d, subject to a semantic-preservation constraint that the crack/non-crack label remains valid. he decision process through which image-condition indicators and dataset-level deficiencies determine transformation type, magnitude, and probability is summarised in Fig. 2.In practical terms, every policy should pass visual audit, thin-crack retention checks, and ablation against a conventional fixed policy. The module is adaptive at policy level; it does not imply that transformations must be optimised online during deployment.

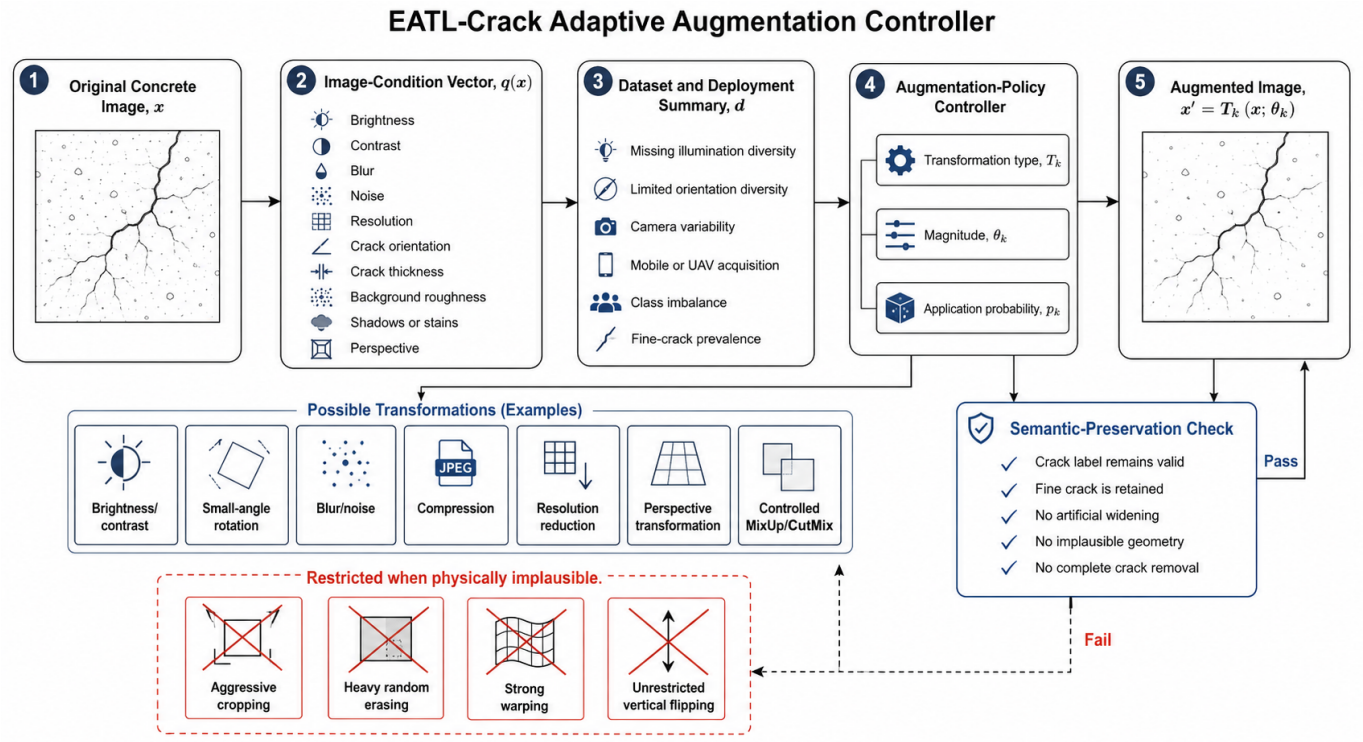


Figure 2: Condition-aware augmentation-policy controller proposed for EATL-Crack. Transformation selection is governed by image-quality indicators, dataset deficiencies, deployment conditions, and crack-semantic preservation constraints.

4.4. Transfer-Learning Module

Candidate ImageNet-pretrained backbones include VGG as a conventional baseline, ResNet for residual optimisation, DenseNet for feature reuse, Inception for multi-scale processing, EfficientNet for compound efficiency scaling, MobileNet for edge devices, and ConvNeXt where resources permit. The same splits, preprocessing, augmentation budget, stopping rules, and metrics should be used for all candidates so that architecture is not confounded with training protocol.

Training proceeds in stages: (1) replace the source classifier with a binary head; (2) freeze the backbone; (3) train the head until validation behaviour stabilises; (4) unfreeze selected upper blocks; (5) fine-tune them with a smaller learning rate; (6) use early stopping and learning-rate scheduling, with class weighting or focal loss when imbalance warrants it [34]; and (7) select the final candidate using predictive, calibration, robustness, and computational criteria. Complete fine-tuning may be tested only when the dataset is sufficiently diverse and overfitting is carefully monitored. The immediate classes are Crack and Non-crack, while the modular head can later be replaced for severity categories.

4.5. Classification and Confidence Module

The classifier produces a logit z and a crack probability

$$P(y = 1 \mid x) = \sigma(z) \tag{2}$$

A threshold converts the probability into a class decision, but 0.5 should not be accepted automatically when false-negative cost, class prevalence, or calibration differs across settings. Recall is emphasised because an undetected crack may delay further assessment. Precision and specificity remain necessary to control unnecessary inspection and alarm burden. Modern neural networks can be poorly calibrated even when classification accuracy is high; temperature scaling is a simple post-training method for aligning confidence with observed correctness on a held-out calibration set [33]. EATL-Crack therefore separates discrimination from confidence. Images in a prespecified uncertainty interval, images with unstable predictions under mild perturbation, or images outside the supported acquisition domain should be referred for manual inspection.

4.6. Explainability Module

Grad-CAM or Grad-CAM++ heatmaps are generated for the crack and non-crack decisions at a selected convolutional layer. Auditors inspect whether activation follows real crack paths and edges or instead concentrates on joints, stains, shadows, textural irregularities, image borders, or background objects. Explanation review should include true positives, true negatives, false positives, false negatives, and low-confidence cases. Where crack masks exist, overlap or pointing-game style checks can complement qualitative review. Disagreement between the prediction and the spatial explanation is a trigger for dataset inspection, policy adjustment, or model rejection. The heatmap is evidence about the spatial basis of the prediction, not proof that the model is causally or structurally correct. The staged optimisation, calibrated decision, manual-referral, and explanation-auditing pathways are integrated as shown in Fig. 3.

4.7. Robustness and Generalisation Module

Robustness evaluation applies controlled noise, blur, brightness and contrast changes, shadows, compression, resolution reduction, and texture perturbation within plausible acquisition ranges. Performance should be reported by corruption type and severity rather than collapsed into a single average. Camera or structure holdout tests reveal source dependence. Cross-dataset testing is stronger than a random within-dataset split because it exposes shifts in illumination, concrete mix and finish, camera characteristics, crack scale, image framing, and background composition. Adaptation on the external test set must be prohibited unless explicitly labelled as a separate domain-adaptation experiment. The proposed evaluation design separates internal discrimination, controlled corruption robustness, source-holdout performance, and external cross-dataset generalisation, as shown in Fig. 4.

4.8. Deployment-Oriented Model Selection

No single metric is sufficient. Selection should include recall, precision, F1, specificity, ROC-AUC, MCC, confusion matrix, expected calibration error, target-device inference time, parameter count, model size, memory, cross-dataset behaviour, corruption stability, and explanation quality. Recall receives elevated weight because missed cracks can delay assessment, but an uncalibrated or confounder-driven model remains unsuitable. The recommendation is a Pareto-efficient candidate satisfying minimum safety and device constraints, not merely the highest-accuracy model.

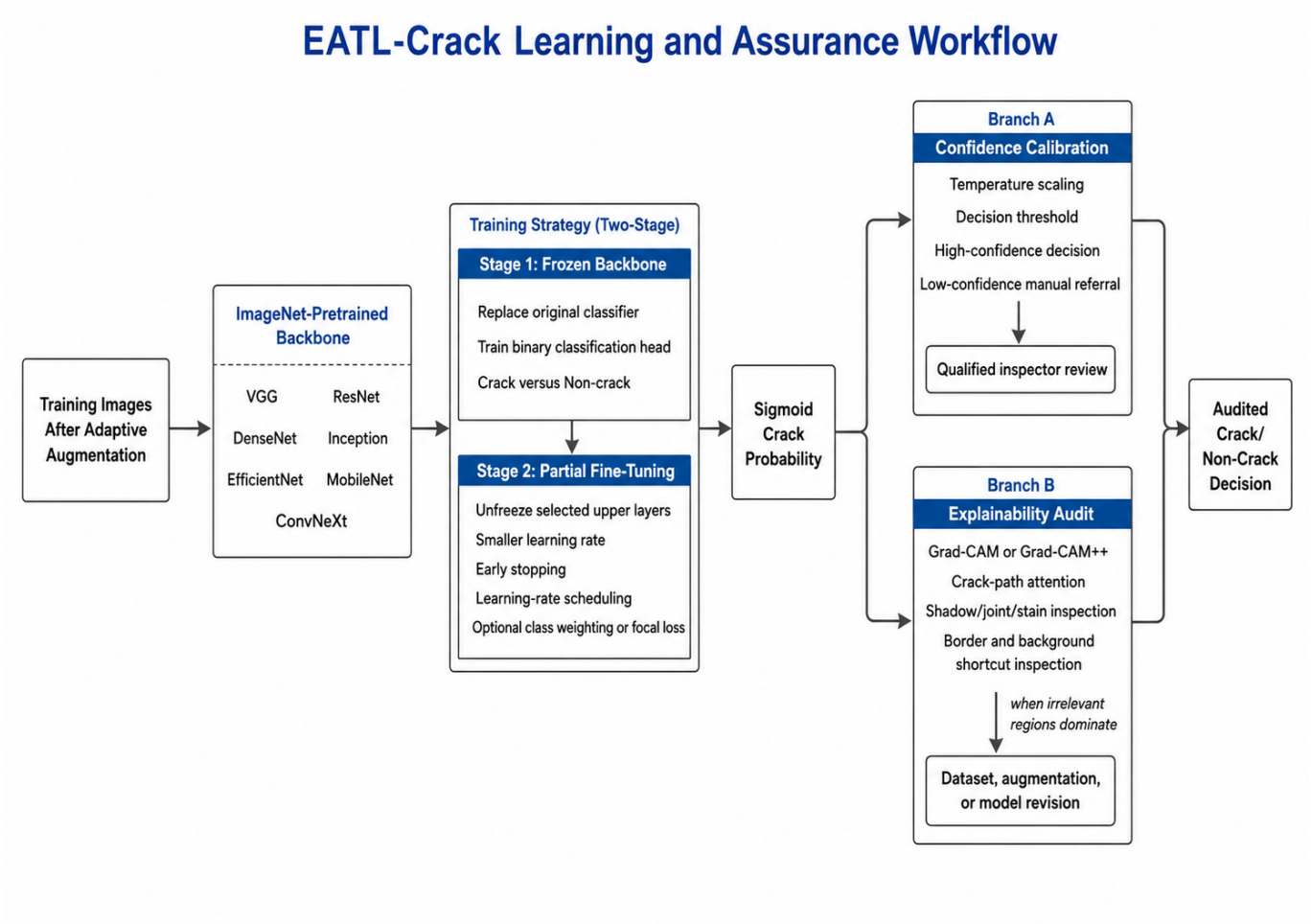


Figure 3: Staged transfer-learning, confidence-calibration, manual-referral, and explanation-auditing workflow of EATL-Crack.

5. Literature-Based Results and Framework Evaluation

The literature supports six EATL-Crack conclusions. Transfer learning is generally more data-efficient than random initialisation for limited crack data [4], [6], [7], [9], [11]. Upper-layer fine-tuning often adapts high-level natural-image features better than a permanently frozen extractor, although excessive unfreezing can increase variance and forgetting. Realistic augmentation reduces overfitting, but transformations that erase or distort cracks can be harmful [9], [24–27]. Lightweight networks may offer a better operational balance, and architecture size alone does not determine performance [11], [14].

Third, realistic augmentation tends to reduce overfitting and broaden exposure to orientation, illumination, and acquisition variation. However, general augmentation literature and crack-specific studies jointly imply that transformation semantics matter [9], [24–27]. Policies that erase the only crack or create implausible geometry can degrade learning even while increasing the nominal image count. Fourth, lightweight architectures can offer a better operational balance than larger networks. Mobile and efficient designs reduce parameter and memory requirements, and crack studies show that larger model size does not guarantee a proportionate benefit across datasets [11], [14]. Device-specific timing remains necessary because theoretical parameter counts do not fully predict latency.

Cross-dataset evaluation is more difficult than random within-dataset testing because illumination, texture, cameras, resolution, crack width, framing, and negative examples shift across sources. Near-saturated internal accuracy therefore does not establish field readiness. Explainability can reveal crack-focused attention or shortcut learning; bridge localisation and feature analyses show its diagnostic value, while saliency sanity checks warn that attractive heatmaps may still be unfaithful [8], [10], [28–32]. Accuracy alone can also hide imbalance, false negatives, poor calibration, and confounding regions.

Table I translates these findings into controls rather than promises of performance. Each component addresses a recurring failure mode reported in prior work, but every response remains testable. The value of the framework is therefore procedural: it requires that data, model, confidence, explanation, robustness, and deployment evidence be

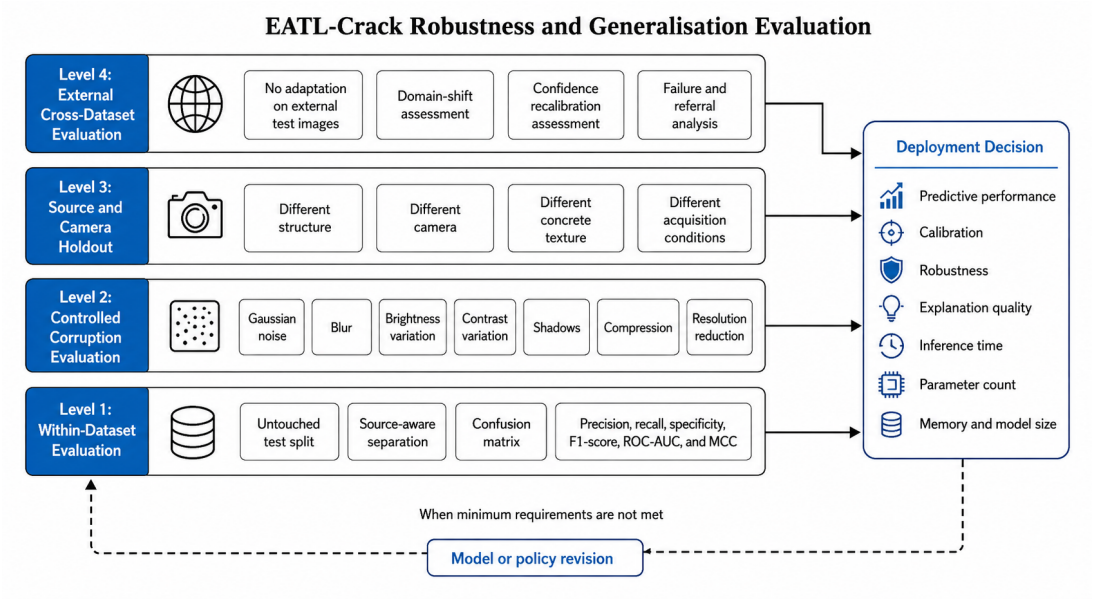


Figure 4: Multi-level robustness and generalisation evaluation protocol proposed for EATL-Crack.

considered together before a recommendation is issued.

Table 1: Comparison of framework components and their literature-supported functions.

Framework component	**Primary function**	**Literature-supported benefit**	**Main limitation**	**Recommended framework response**
Data preparation	Create leakage-resistant partitions and document sources.	More credible generalisation estimates and traceable errors.	Structure identifiers or acquisition metadata may be absent.	Use perceptual duplicate checks, source-image grouping, and transparent assumptions.
Condition assessment	Characterise brightness, blur, noise, scale, texture, shadow, and perspective.	Makes augmentation and subgroup analysis responsive to observed variation.	Quality indicators may not correspond perfectly to field difficulty.	Validate indicators against expert review and failure cases.
Adaptive augmentation	Select realistic transformations, magnitudes, and probabilities.	Improves diversity while protecting crack semantics.	Controller complexity and risk of unrealistic samples.	Apply bounded policies, visual audit, thin-crack retention tests, and ablation.
Staged transfer learning	Train the head, then adapt selected upper layers.	Combines pretrained efficiency with domain-specific feature adaptation.	Overfitting or catastrophic forgetting with excessive unfreezing.	Use smaller learning rates, early stopping, and layer-wise comparisons.
Confidence calibration	Align reported probability with observed correctness.	Supports thresholds, uncertainty referral, and risk-aware use.	Calibration may shift across cameras or structures.	Calibrate on held-out data and reassess after domain change.
Explainability audit	Inspect spatial evidence using Grad-CAM/Grad-CAM++ and alternatives.	Reveals crack-focused attention and possible shortcuts.	Post-hoc maps may be coarse, unstable, or unfaithful.	Use sanity checks, error-stratified review, and mask-based validation where available.
Robustness and deployment	Test corruptions, external sources, latency, size, and memory.	Connects laboratory discrimination to operational suitability.	External data and representative hardware may be limited.	Report constraints explicitly and defer deployment until prospective validation.

Table 2: Recommended evaluation protocol for the EATL-Crack framework.

Evaluation category	**Recommended measure**	**Purpose**	**Interpretation**
Overall discrimination	Accuracy	Summarise correct classifications.	Report with class distribution; never use alone.
Positive prediction	Precision	Measure how often predicted cracks are genuine.	Low precision implies excessive false alarms.
Crack sensitivity	Recall	Measure the proportion of real cracks detected.	Primary safety-oriented measure; inspect false negatives.
Negative-class control	Specificity	Measure correct rejection of non-crack images.	Low specificity indicates unnecessary referrals or alarms.
Balanced classification	F1-score	Combine precision and recall.	Useful when crack and non-crack error costs both matter.
Threshold-independent ranking	ROC-AUC	Assess ordering across decision thresholds.	Complement with precision–recall behaviour under imbalance.
Imbalance-aware correlation	MCC	Summarise all confusion-matrix cells.	Values near zero indicate weak association despite possible accuracy inflation.
Error structure	Confusion matrix	Show false-positive and false-negative counts.	Stratify by source and image condition.
Confidence quality	Expected calibration error	Compare confidence with empirical correctness.	Lower is better; verify after domain shift.
Operational efficiency	Inference time, parameters, model size, memory	Assess suitability for target hardware.	Measure on the actual smartphone, drone, edge, or server platform.
Generalisation	Cross-dataset and source-holdout performance	Expose dependence on camera, structure, texture, and acquisition.	A material drop signals domain shift and need for adaptation or referral.
Explanation quality	Heatmap inspection and, where possible, overlap with crack regions	Audit whether decisions rely on plausible damage areas.	Reject models dominated by borders, shadows, joints, stains, or unrelated texture.

6. Discussion

EATL-Crack is more comprehensive than a backbone comparison because it treats classification as one part of an inspection chain. A ranking from one random split cannot show whether duplicate patches leaked, thin cracks survived augmentation, confidence supports referral, attention follows damage, or performance transfers to another camera. Requiring evidence at each stage makes optimism and uncertainty auditable.

Adaptive augmentation is central because transformations are useful only when they address missing variation without changing the label. Brightness changes may cover illumination; large crops may invalidate hairline-crack images; perspective changes fit oblique acquisition but not every fixed rig. The controller therefore represents a documented hypothesis about deployment variation and should be compared with fixed and no-augmentation baselines, with generated-sample audits and subgroup analysis.

Performance–efficiency trade-offs depend on the application. A large backbone may provide a small predictive gain yet exceed memory, energy, or latency limits on drones and smartphones. MobileNet and EfficientNet are plausible edge candidates; ResNet, DenseNet, or ConvNeXt may suit server inference. Because quantisation and hardware acceleration can change rankings, latency and memory must be measured on the target device. A smaller, calibrated, interpretable model may be preferable to a larger model with marginally higher accuracy.

Explainability supports decision making by showing whether attention follows crack paths or confounders such as joints and shadows. Grad-CAM is coarse, Grad-CAM++ remains gradient based, and both depend on layer choice. Integrated gradients and SHAP add complementary views but also depend on baselines and approximations. Explanations should support error analysis and human–AI dialogue, not replace validation or engineering judgement.

Cross-dataset validation is essential because concrete mix, finish, weathering, moisture, coatings, optics, distance, and negative-class composition vary. Bias arises from uniform backgrounds, repeated crops, laboratory lighting, or border conventions. Patch-level random division is especially risky because nearly identical texture can cross sets. Source-level

separation and duplicate detection are therefore prerequisites for credible evaluation.

Image-level classification indicates likely crack presence but does not measure width, length, continuity, orientation, or severity. Thin cracks may disappear after resizing, while shadows, joints, stains, rough aggregate, moisture, and other defects can mimic lines. Segmentation and calibrated measurement are required when geometry informs maintenance [12], [14]. Low-confidence or contradictory cases should be routed to qualified inspectors and interpreted with structural context.

Lightweight models enable screening on smartphones, drones, embedded systems, and edge platforms, reducing data transfer and supporting rapid field feedback. They also introduce device-specific issues such as compression effects, battery use, thermal throttling, and camera calibration. EATL-Crack is therefore an implementation blueprint that still requires prospective, site-specific validation before operational use.

7. Limitations and Future Research

This framework depends on published evidence and introduces no new images, models, or measurements. Studies vary in resolution, crack definition, class balance, split protocol, augmentation, optimisation, and metrics, limiting direct comparison. Publication bias may underrepresent failed transfer, harmful augmentation, calibration problems, and deployment limitations. Open-access criteria also shape coverage. Post-hoc explanations remain imperfect, and binary classification does not measure crack geometry or severity.

Future work should implement EATL-Crack with pre-registered source-aware splits and external validation, then extend it to severity classification, segmentation, width measurement, and detection. Vision transformers, hybrid backbones, self-supervised learning, domain adaptation, few-shot learning, and synthetic data require controlled comparison. Multimodal and temporal sensing could support progression assessment. Drone acquisition, edge deployment, and human–AI collaboration should be evaluated prospectively on real assets, representative hardware, inspector workload, referral thresholds, and maintenance decisions.

8. Conclusion

Automated concrete surface crack classification can support more scalable infrastructure inspection, but a high score on a homogeneous image split is insufficient evidence of safe field use. This paper proposed EATL-Crack, an Explainable Adaptive Transfer Learning Concrete Crack Classification Framework that integrates source-aware data preparation, image-condition assessment, adaptive augmentation, staged transfer learning, calibrated confidence, Grad-CAM-based auditing, robustness and cross-dataset testing, and deployment-oriented model selection. The literature supports the individual value of transfer learning, realistic augmentation, controlled fine-tuning, uncertainty assessment, and visual explanation, while also showing that model behaviour is sensitive to dataset design and domain shift. EATL-Crack therefore offers a transparent blueprint rather than a claim of experimental superiority. Its practical significance lies in directing future systems toward crack-sensitive, confidence-aware, inspectable, and computationally feasible decisions that can complement human expertise in infrastructure maintenance.

Author Contributions

Vinay Deeti (V.D): Conceptualization, Methodology, Investigation, Writing – original draft; Shilpa Bade-Gite (S.B.G): Methodology, Validation, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

This study did not generate or analyze any new datasets; the framework is synthesized entirely from previously published, peer-reviewed, cited works. Enquiries about the synthesis can be directed to the corresponding author, V.D.

Declaration of Generative AI and AI-Assisted Technologies in the Writing Process

During the preparation of this work the authors did not use any generative AI or AI-assisted technologies.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] H. S. Munawar, A. W. A. Hammad, A. Haddad, C. A. P. Soares, and S. T. Waller, "Image-Based Crack Detection Methods: A Review," Infrastructures, vol. 6, no. 8, Art. no. 115, 2021, doi: 10.3390/infrastructures6080115.

[2] M. A.-M. Khan, S.-H. Kee, A.-S. K. Pathan, and A.-A. Nahid, "Image Processing Techniques for Concrete Crack Detection: A Scientometrics Literature Review," Remote Sens., vol. 15, no. 9, Art. no. 2400, 2023, doi: 10.3390/rs15092400.

[3] S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," Constr. Build. Mater., vol. 186, pp. 1031–1045, 2018, doi: 10.1016/j.conbuildmat.2018.08.011.

[4] W. R. L. da Silva and D. S. de Lucena, "Concrete Cracks Detection Based on Deep Learning Image Classification," Proceedings, vol. 2, no. 8, Art. no. 489, 2018, doi: 10.3390/ICEM18-05387.

[5] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," Autom. Constr., vol. 99, pp. 52–58, 2019, doi: 10.1016/j.autcon.2018.11.028.

[6] Q. Yang, W. Shi, J. Chen, and W. Lin, "Deep convolution neural network-based transfer learning method for civil infrastructure crack detection," Autom. Constr., vol. 116, Art. no. 103199, 2020, doi: 10.1016/j.autcon.2020.103199.

[7] R. Rajadurai and S.-T. Kang, "Automated Vision-Based Crack Detection on Concrete Surfaces Using Deep Learning," Appl. Sci., vol. 11, no. 11, Art. no. 5229, 2021, doi: 10.3390/app11115229.

[8] U. H. Billah, H. M. La, and A. Tavakkoli, "Deep Learning-Based Feature Silencing for Accurate Concrete Crack Detection," Sensors, vol. 20, no. 16, Art. no. 4403, 2020, doi: 10.3390/s20164403.

[9] M. M. Islam, M. B. Hossain, M. N. Akhtar, M. A. Moni, and K. F. Hasan, "CNN Based on Transfer Learning Models Using Data Augmentation and Transformation for Detection of Concrete Crack," Algorithms, vol. 15, no. 8, Art. no. 287, 2022, doi: 10.3390/a15080287.

[10] H. Zoubir, M. Rguig, M. El Aroussi, A. Chehri, R. Saadane, and G. Jeon, "Concrete Bridge Defects Identification and Localization Based on Classification Deep Convolutional Neural Networks and Transfer Learning," Remote Sens., vol. 14, no. 19, Art. no. 4882, 2022, doi: 10.3390/rs14194882.

[11] R. E. Philip, A. D. Andrushia, A. Nammalvar, B. G. A. Gurupatham, and K. Roy, "A Comparative Study on Crack Detection in Concrete Walls Using Transfer Learning Techniques," J. Compos. Sci., vol. 7, no. 4, Art. no. 169, 2023, doi: 10.3390/jcs7040169.

[12] M. Iraniparast, S. Ranjbar, M. Rahai, and F. Moghadas Nejad, "Surface concrete cracks detection and segmentation using transfer learning and multi-resolution image processing," Structures, vol. 54, pp. 386–398, 2023, doi: 10.1016/j.istruc.2023.05.062.

[13] Y. Zhang, Y.-Q. Ni, X. Jia, and Y.-W. Wang, "Identification of concrete surface damage based on probabilistic deep learning of images," Autom. Constr., vol. 156, Art. no. 105141, 2023, doi: 10.1016/j.autcon.2023.105141.

[14] C. Su and W. Wang, "Concrete Cracks Detection Using Convolutional Neural Network Based on Transfer Learning," Math. Probl. Eng., vol. 2020, Art. no. 7240129, 2020, doi: 10.1155/2020/7240129.

[15] M. A. Nyathi, J. Bai, and I. D. Wilson, "Deep Learning for Concrete Crack Detection and Measurement," Metrology, vol. 4, no. 1, pp. 66–81, 2024, doi: 10.3390/metrology4010005.

[16] S. Dorafshan, R. J. Thomas, and M. Maguire, "SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks," Data Brief, vol. 21, pp. 1664–1668, 2018, doi: 10.1016/j.dib.2018.11.015.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[18] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2017, pp. 4700–4708, doi: 10.1109/CVPR.2017.243.

[19] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in Proc. Int. Conf. Learn. Represent. (ICLR), 2015.

[20] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[22] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in Proc. 36th Int. Conf. Mach. Learn. (ICML), vol. 97, 2019, pp. 6105–6114.

[23] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2022, pp. 11976–11986, doi: 10.1109/CVPR52688.2022.01167.

[24] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random Erasing Data Augmentation," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 7, 2020, pp. 13001–13008, doi: 10.1609/aaai.v34i07.7000.

[25] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning Augmentation Strategies From Data," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2019, pp. 113–123, doi: 10.1109/CVPR.2019.00020.

[26] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond Empirical Risk Minimization," in Proc. Int. Conf. Learn. Represent. (ICLR), 2018.

[27] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features," in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), 2019, pp. 6023–6032, doi: 10.1109/ICCV.2019.00612.

[28] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations From Deep Networks via Gradient-Based Localization," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.

[29] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks," in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), 2018, pp. 839–847, doi: 10.1109/WACV.2018.00097.

[30] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks," in Proc. 34th Int. Conf. Mach. Learn. (ICML), vol. 70, 2017, pp. 3319–3328.

[31] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 4765–4774.

[32] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity Checks for Saliency Maps," in Adv. Neural Inf. Process. Syst., vol. 31, 2018, pp. 9505–9515.

[33] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in Proc. 34th Int. Conf. Mach. Learn. (ICML), vol. 70, 2017, pp. 1321–1330.

[34] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.324.