

TRACE-Pave: A Neural Perception and Auditable Decision Framework for Explainable Road Crack Assessment

Tayyab Rehman¹, Giovanni De Gasperis¹

¹DISIM, Università degli Studi dell'Aquila, L'Aquila, Italy

Abstract

Automated crack analysis has largely solved the perception problem while leaving the decision problem untouched. A class label and a mask do not tell a highway engineer how wide a crack is, how severe it is, which treatment applies, or which rule produced that answer, and none of this can be audited from a model score. TRACE-Pave draws an explicit authority boundary through the pipeline. Learned computation stops at perception, where the system classifies, detects, and segments. Every quantity an engineer must defend is derived below that boundary by deterministic rules whose inputs, thresholds, and intermediate values are recorded: width, extent, severity, a prototype crack-condition score, and a maintenance action. A task-decoupled stage refines segmentation with classification and detection frozen, so a better mask cannot silently revise a class call the engineer has already accepted. The reported configuration reaches 95.02% classification macro F1, 95.50% detection F1, and 83.40% Dice, while the decision layer costs 0.61 of 37.67 ms per image, which makes audibility close to free. An audit of the data pipeline is also reported, one that disqualifies these figures as generalisation evidence, together with three cases in which a convincing saliency map accompanies a confident error.

Keywords

Road crack assessment, Explainable artificial intelligence, Auditable decision support, Task-decoupled segmentation, Human oversight

1. Introduction

Road networks are among the most capital-intensive assets under public stewardship, and their surfaces degrade continuously under traffic loading, thermal cycling, and moisture ingress. The timing of intervention drives lifecycle cost: sealing a narrow crack early is far cheaper than the reconstruction required once damage reaches the base layer, so an assessment error is not merely a misclassification but a budgetary and safety decision made wrongly. Conventional condition surveys classify distress by type, severity, and extent, using procedures such as ASTM D6433 [1], but they are slow, costly, and subject to inter-rater variability.

Three distress patterns define the classes used here. *Longitudinal* cracks run parallel to the direction of travel, often along construction joints or wheel paths. *Transverse* cracks run across the carriageway and are commonly thermal in origin. *Alligator* cracking is an interconnected network of polygonal cells resembling reptile skin, and it signals fatigue failure of the base or subgrade. The distinction carries engineering weight, since alligator cracking implies a deeper intervention than a longitudinal seal of the same width, which is why crack type enters the maintenance rules of Section 3.5 rather than serving as a label alone. Neural systems can automate crack classification, detection, and segmentation [2, 3], but prediction accuracy alone does not establish whether a maintenance recommendation is physically justified or traceable. An engineer must also understand how the observed damage was measured and which rule converted those measurements into an action. A condition survey does not terminate at a class label, a box, or a mask; it opens a decision chain that may end in a maintenance action with legal, safety, and budgetary consequences.

Explainable artificial intelligence (XAI) addresses aspects of this problem, but different forms of explanation answer different questions. Gradient-weighted class activation mapping (Grad-CAM)

XAI.it 2026 - 6th Workshop on Explainable AI, Generative and Agentic Systems: Trust, Transparency, and Human Oversight, co-located with AIXIA 2026, October 6-9, 2026, Perugia, Italy

✉ tayyab.rehman@graduate.univaq.it (T. Rehman); giovanni.degasperis@univaq.it (G. D. Gasperis)

id 0000-0002-1554-2131 (T. Rehman); 0000-0001-9521-4711 (G. D. Gasperis)



© 2026 Copyright © 2026 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

highlights regions associated with a class prediction [4]; it does not explain how a segmentation mask becomes a severity rating. Stakeholder-oriented explanation therefore requires both a bounded interpretation of neural diagnostics and an inspectable decision process [5, 6]. Visual plausibility must not be confused with empirically evaluated explanation quality [7].

Regulation points the same way. The EU Artificial Intelligence Act requires high-risk systems to be transparent enough for deployers to interpret and appropriately use their output, and to support human oversight [8]. Road-maintenance decision support is not currently designated high-risk, and no assessment under the Act is claimed here; the principles it codifies nonetheless describe what an infrastructure workflow already needs. Letting a generative model produce decision-relevant quantities without constraint runs against them, since such models can emit plausible but unsupported severity ratings [9].

Problem statement. Given a pavement image, an assessment system must return not only a crack type and a mask but a defensible maintenance recommendation. Three requirements follow. The measured quantities must be traceable to the mask and the scale that produced them. The rule that selected a treatment must be identifiable after the fact. The engineer must be able to contest any single step without discarding the whole output. Current systems satisfy none of these, because their outputs terminate at perception. Strong individual results in crack detection and segmentation are now common, but no published system appears to carry perception output through to an auditable, contestable engineering decision. That gap, rather than benchmark accuracy, is what this work addresses.

TRACE-Pave separates learned perception from deterministic decision support. In *Phase A*, a multi-task baseline predicts crack type, boxes, and masks. In *Phase B*, the CTAG-Seg module refines segmentation while the classification and detection branches remain frozen. The resulting mask is converted into width, contour-based length, coverage, and density, followed by severity, a prototype crack-condition score (pCCS), and a maintenance recommendation. Intermediate values and rule inputs are retained in JavaScript Object Notation (JSON) records. Grad-CAM is restricted to classification inspection, and final decision authority remains with an engineer.

The contribution is an integration of established techniques rather than a new backbone or attribution algorithm, stated openly here because several components are well known and presenting them as novel would misdescribe the work. Three contributions follow from the composition:

1. **An authority boundary between learned and deterministic computation.** Every quantity an engineer must defend—width, length, coverage, severity, score, and treatment—is derived from the mask by an explicit rule with recorded inputs. No learned component writes a decision value.
2. **Task-decoupled segmentation refinement.** CTAG-Seg raises resolution using crack-type conditioning, multi-scale context, and boundary-aware training with classification and detection frozen, so that improving a mask cannot silently alter an already-accepted class call.
3. **A configuration-level evaluation reported with its own limits,** comparing phases at selected and matched thresholds, presenting high-confidence failures, and reporting a pipeline audit that invalidates the numbers as generalisation evidence.

That last point sets the scope of everything that follows. Oversampling preceded partitioning in the reported checkpoint, so source images overlap between training and test; the results characterise this prototype and do not establish leakage-free generalisation. They are reported because the architecture and the matched-threshold behaviour remain informative, and because a disclosed measurement problem is more useful to a reader than an undisclosed one.

Section 2 positions TRACE-Pave against prior pavement-analysis and XAI work. Section 3 presents the methodology and the auditable workflow. Section 4 reports the experimental setup and results. Section 5 discusses implications and limitations, and Section 6 concludes.

2. Related Work and Positioning

Prior work is reviewed along three axes that together define the gap addressed here: pavement perception and severity assessment, conditional segmentation for thin structures, and the evaluation of post-hoc explanation. Each axis is strong on its own terms; what is missing is the connection between them, which Section 2.4 makes explicit.

2.1. Pavement perception

Ashraf et al. [2] combine multi-scale feature aggregation and transformer attention for classification, detection, and segmentation, providing the architectural reference for Phase A. Yang et al. [3] address thin-crack segmentation through feature pyramids and hierarchical boosting. More recent approaches adapt the Segment Anything Model through low-rank fine-tuning [10] or reduce annotation requirements through weakly supervised adversarial learning [11]. These directions advance perception; connecting their outputs to reproducible engineering decisions remains a separate design problem.

Work that does reach a severity judgement is closer to the present concern. Matarneh et al. [12] classify cracks and assess severity from Hough-transform features, reporting 82.96% classification precision and 71.46% severity precision. That pipeline establishes severity as a target worth predicting, but it operates without pixel-level segmentation and without a recorded path from measurement to recommendation, so the severity call cannot be contested at the level of the quantity that produced it. TRACE-Pave takes the opposite route: severity is not predicted at all but derived from a measured mask by an explicit rule, which trades learned flexibility for inspectability. The review by Huang et al. [13] gives a broader account of machine-vision crack detection and reports that the field has not converged on severity-mapping protocols or explainability criteria that would make automated outputs usable inside pavement-management systems.

2.2. Conditional segmentation

Feature-wise linear modulation (FiLM) conditions features on auxiliary information [14], while atrous spatial pyramid pooling (ASPP) captures context at several dilation rates [15]. TRACE-Pave uses class information to condition segmentation and higher-resolution features to retain thin structures. The imbalance is severe rather than incidental: crack pixels typically occupy under 2% of an image, a ratio at which pixel-wise cross-entropy is nearly minimised by predicting all background. Dice and Tversky objectives address this directly, and a boundary-focused term emphasises crack edges, where a low-resolution decoder loses the most detail [16]. These choices are motivated by complementary functions; their individual benefits have not been isolated experimentally.

2.3. Explanation and decision support

Grad-CAM and Grad-CAM++ provide class-discriminative visual diagnostics [4, 17]. Perturbation-based approaches, including randomized input sampling for explanation (RISE) and remove-and-retrain evaluation (ROAR), illustrate the need to test attribution rather than rely on heatmaps alone [18, 19]. TRACE-Pave does not claim to solve neural explanation faithfulness; it adds procedural auditability downstream of perception. Within pavement analysis specifically, Gulfam et al. [20] apply attribution analysis to a lightweight crack classifier and confirm that it attends to cracked regions. This is the closest published pairing of pavement perception with explanation, and it illustrates the limit this work seeks to pass: confirming that a model looks at the crack says nothing about whether the resulting maintenance decision is defensible. Concept bottleneck models [21] offer a different route, routing prediction through human-meaningful intermediate concepts, though their use for dense prediction such as crack segmentation remains limited.

This separation also limits the authority of generative reporting. Language models can produce unsupported statements [9]; consequently, an optional narrator receives completed records and has no write authority over decision values. Human-oversight and transparency principles in the EU Artificial

Table 1

Positioning against representative prior work. *Expl.*: an explanation mechanism is provided. *Eval.*: explanation quality is evaluated rather than assumed. *Decision*: an inspectable mapping from model output to an engineering recommendation. *Prov.*: intermediate values are recorded for later inspection.

Work	Expl.	Eval.	Decision	Prov.	Limitation for auditable road assessment
Ashraf et al. [2]	×	×	×	×	Aggregate perception metrics; no mask-to-decision chain
Yang et al. [3]	×	×	×	×	Binary masks without typed decision mapping
Arya et al. [22]	×	×	×	×	Box supervision; no segmentation or severity output
Matarneh et al. [12]	×	×	Partial	×	Severity predicted, not derived; no pixel-level evidence
Guo et al. [10]	×	×	×	×	Foundation-model adaptation; perception stops at the mask
Ma et al. [11]	×	×	×	×	Lower annotation cost; no severity or scoring rules
Gulfam et al. [20]	⊠	×	×	×	Attribution confirms attention only; no decision protocol
Selvaraju et al. [4]	⊠	×	×	×	Coarse attribution; no downstream engineering link
Chattopadhyay et al. [17]	⊠	×	×	×	Sharper maps, but no domain decision protocol
Hooker et al. [19]	×	⊠	×	×	Faithfulness protocol; not an end-to-end system
Koh et al. [21]	⊠	×	×	×	Interpretable concepts; limited for dense prediction
TRACE-Pave	⊠	×	⊠	⊠	Only entry closing mask → geometry → severity → treatment with recorded inputs; attribution remains qualitative and the historical split overlaps

Intelligence Act motivate this governance boundary [8], without implying that the prototype has been legally classified or certified under the Act.

2.4. Positioning summary

The backbone, the detection head, the decoder, FiLM, ASPP, the Tversky objective, and Grad-CAM are all taken from the work cited above and used essentially as published; what this paper adds is the authority boundary, the task-decoupled refinement stage, and the recorded mask-to-decision chain. Table 1 places the result against representative prior systems, and reading down its *Decision* and *Prov.* columns exposes two systematic gaps.

Read as a whole, the table shows that capability and accountability have advanced separately. Every entry above the rule performs its own task competently; none records how its output would become a decision. TRACE-Pave is the only entry carrying both a decision mapping and stored provenance, and it is also the only entry whose own row names a measurement limitation, which is the standard that ought to be applied to the others. Two systematic gaps follow.

The first is a coupling gap. The works that supply an explanation mechanism and the works that produce engineering-relevant outputs are largely disjoint sets. Attribution methods are developed and evaluated on classification benchmarks, while pavement systems that reach a severity or treatment output do so with no inspectable link back to perception. Perception, saliency, and downstream rules serve genuinely different purposes: a better mask does not validate maintenance logic, and a deterministic rule does not make a neural prediction faithful. Treating these as one undifferentiated pipeline is what allows an unexamined class call to propagate silently into a recommendation.

The second is an evaluation gap. Only the faithfulness literature tests explanations at all, and it does so on generic vision tasks rather than domain pipelines; applied pavement work reports that the model attends to cracked regions and stops. TRACE-Pave closes the first gap by construction and addresses the second only partially. The decision chain is auditable step by step because its inputs, equations, and thresholds are recorded, but the attribution component is still evaluated on three qualitative examples, which is why the TRACE-Pave row carries that limitation explicitly rather than claiming otherwise.

3. Methodology

This section describes the pipeline in the order a road image travels through it. Section 3.1 states the data sources and, critically, the partition protocol. Sections 3.2 and 3.3 cover the two training stages.

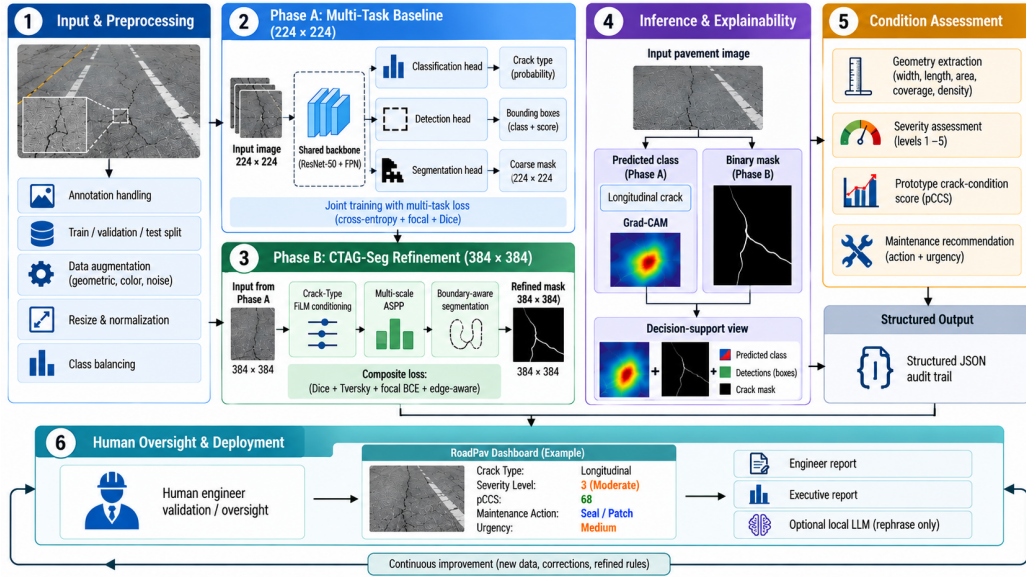


Figure 1: TRACE-Pave workflow connecting data preparation, two-phase neural perception, auditable condition assessment, and engineer-governed reporting.

Table 2

Source counts, supervision, and split roles. Counts refer to the locally used files or expanded rows, not necessarily the complete public releases.

Source	Count	Supervision	Role
RCD-IIUM	1,877	Typed labels, boxes, polygon masks	Source of expanded typed rows
RDD2022	2,000	Typed labels and boxes	No segmentation loss or metrics
CRACK500	3,868	Binary masks	Training only; classification loss masked
Expanded typed pool	10,000	Three crack classes	Historical 7,000/1,500/1,500 row split

Section 3.4 covers inference and classifier diagnostics, and Section 3.5 covers the deterministic rules that convert a mask into a recommendation.

Figure 1 connects data preparation, two training phases, task-decoupled inference, condition assessment, and engineer-governed reporting. The source-first split shown is the corrected protocol; the historical results in this paper precede leakage-free retraining. The optional language model can narrate completed JSON fields but cannot change the authoritative record.

3.1. Data and annotation provenance

The typed sources comprise 1,877 locally available RCD-IIUM images [23] and a capped subset of 2,000 RDD2022 files [22]. RDD2022 categories D00, D10, and D20 map to longitudinal, transverse, and alligator cracking, respectively. Alligator cracking denotes an interconnected, block-like crack pattern. CRACK500 supplies 3,868 binary-mask training samples [3]. Table 2 separates original typed images, expanded rows, and mask supervision; bounding boxes are never treated as segmentation masks.

The historical pipeline expanded the 3,877 typed source images to 10,000 rows: 3,200 alligator, 3,600 longitudinal, and 3,200 transverse. Oversampling preceded the seeded 70%/15%/15% row split, producing 7,000 training, 1,500 validation, and 1,500 test rows. The source audit found 975 train–validation, 1,004 train–test, and 458 validation–test overlaps. Thus, test rows are not a source-independent holdout and the resulting metrics may be optimistic. CRACK500 was appended only to training, with classification loss masked. Online augmentation uses random crops, rotations, flips, colour changes, and noise.

The corrected loader partitions canonical source identifiers before training-only balancing and augmentation. However, the reported checkpoint has not been retrained using that protocol. The

correction therefore does not remove leakage from the results reported here. Segmentation losses and metrics use only genuine-mask samples, including 877 rows in the typed test partition.

Source resolutions also differ. RCD-IIUM is uniformly 640×640 ; RDD2022 ranges from 512×512 to 4040×2035 (median 640×640); and CRACK500 ranges from 640×360 to 3264×2448 (median 640×360). The retained study records do not support a harmonised per-image comparison of camera type, acquisition angle, lighting, or road-surface appearance. These acquisition factors remain uncontrolled domain variation, rather than being assumed equivalent across datasets.

3.2. Phase A: multi-task baseline

A ResNet-50 encoder and feature pyramid network (FPN) [24] produce features at strides $\{4, 8, 16, 32\}$, which feed transformer attention and three task-specific heads. Following the reference architecture [2], the classification branch uses a ResNet-34 head. Attended features F_{att} yield three class probabilities through global average pooling (GAP):

$$\hat{\mathbf{y}} = \text{softmax}(W_c \text{GAP}(F_{\text{att}}) + b_c), \quad \mathcal{L}_{\text{CE}} = - \sum_{c=1}^3 y_c \log \hat{y}_c,$$

where $\mathbf{y}$ is the one-hot class target. A RetinaNet-style detection branch [25] combines focal classification loss with smooth- L_1 box regression:

$$\mathcal{L}_{\text{det}} = \mathcal{L}_{\text{focal}}(\hat{p}, p^*) + \lambda_{\text{box}} \mathcal{L}_{\text{smooth-}L_1}(\hat{b}, b^*).$$

Here, $\hat{p}, p^*$ denote detection predictions and targets, $\hat{b}, b^*$ denote boxes, and λ_{box} weights box regression. The U-Net decoder [26] predicts masks at 224×224 , with soft Dice loss

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2 \sum_i p_i g_i + \epsilon}{\sum_i p_i + \sum_i g_i + \epsilon}, \quad \mathcal{L}_A = \mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{det}} + \mathcal{L}_{\text{Dice}},$$

where $p_i \in [0, 1]$, $g_i \in \{0, 1\}$, and $\epsilon > 0$ stabilises division. Cross-entropy supervises crack type, focal loss addresses detection imbalance, and Dice emphasises foreground overlap. Each term is applied only when its corresponding annotation is available. Phase A remains responsible for classification and detection at deployment.

3.3. Phase B: task-decoupled CTAG-Seg

CTAG-Seg operates at 384×384 to retain thin-crack detail while freezing Phase A. FiLM conditions an intermediate feature map h using per-channel scale and shift:

$$h' = h \odot (1 + \gamma(\mathbf{c})) + \beta(\mathbf{c}), \quad [\gamma(\mathbf{c}); \beta(\mathbf{c})] = \text{MLP}(\mathbf{c}).$$

The multilayer perceptron (MLP) has two 64-unit hidden layers with rectified linear unit (ReLU) activations. Typed training samples use their ground-truth class encoding $\mathbf{c}$; CRACK500 and inference use predicted class probabilities. This training–inference conditioning mismatch is not separately evaluated.

ASPP combines a 1×1 convolution, atrous convolutions at rates $\{6, 12, 18\}$, and a spatially broadcast GAP branch; concatenation and a 1×1 projection produce 256 channels. The rates are those of the original DeepLab configuration and were not re-tuned. At 384×384 input they span receptive fields from roughly the width of a hairline crack to that of an alligator cell, which is the range on which the crack-type distinction depends. These branches supply complementary context without changing the frozen classification and detection outputs.

The Phase B training objective is:

$$\mathcal{L}_B = 0.35 \mathcal{L}_{\text{Dice}} + 0.30 \mathcal{L}_{\text{Tversky}} + 0.20 \mathcal{L}_{\text{focalBCE}} + 0.15 \mathcal{L}_{\text{boundary}}. \quad (1)$$

Binary cross-entropy (BCE) receives focal weighting to emphasise difficult pixels. The Tversky term uses $\alpha_T = 0.3$ for false positives and $\beta_T = 0.7$ for false negatives:

$$\mathcal{L}_{\text{Tversky}} = 1 - \frac{\sum_i p_i g_i + \epsilon}{\sum_i p_i g_i + 0.3 \sum_i p_i (1 - g_i) + 0.7 \sum_i (1 - p_i) g_i + \epsilon}.$$

Table 3

Prototype severity intervals and deductions. Widths are in millimetres; deductions are score points. Intervals are lower-inclusive and upper-exclusive.

Level ℓ	Severity	Width interval $w_{\max}$ (mm)	$d(\ell)$
1	Hairline	$0.0 \leq w_{\max} < 0.5$	5
2	Low	$0.5 \leq w_{\max} < 2.0$	12
3	Moderate	$2.0 \leq w_{\max} < 5.0$	22
4	High	$5.0 \leq w_{\max} < 12.0$	38
5	Severe	$w_{\max} \geq 12.0$	55

The boundary term applies focal BCE to a band around the ground-truth mask M^* , $B = \text{dilate}(M^*, k) - \text{erode}(M^*, k)$, with structuring element k . The weighting prioritises region overlap and missed crack pixels while retaining boundary supervision. It is a design choice, not an ablation-supported optimum. Phase B introduces 4.10 million trainable parameters.

3.4. Inference and classification diagnostics

The 224-pixel Phase A path returns class probabilities and detections; the 384-pixel CTAG-Seg path returns mask probabilities. Horizontal-flip test-time augmentation (TTA) averages the original and spatially restored flipped predictions:

$$P = \frac{1}{2}(P_{\text{orig}} + \text{flip}^{-1}(P_{\text{flip}})).$$

Validation Dice selects a threshold from $\{0.30, 0.35, \dots, 0.85\}$, yielding $t_A^* = 0.75$ and $t_B^* = 0.35$. These are the *selected operating points*. Morphological opening and closing precede connected-component filtering:

$$M = \text{CCFilter}(\text{close}(\text{open}(P > t^*)), A_{\min}),$$

where $A_{\min}$ is the minimum retained component area. Grad-CAM is computed for class score y^c from feature maps A^k :

$$I_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k\right), \quad \alpha_k^c = \frac{1}{Z} \sum_{i,j} \frac{\partial y^c}{\partial A_{ij}^k},$$

with Z the feature-map area. This map is a classification diagnostic, not a causal explanation of segmentation or a validation of the decision layer.

3.5. Auditable condition assessment

For a post-processed mask $M \in \{0, 1\}^{H \times W}$, an engineer supplies the scale s in millimetres per pixel. With Euclidean distance transform D , the maximum-width proxy and contour-based length are

$$w_{\max} = 2s \max_x D(x) \quad [\text{mm}], \quad L = \frac{s}{1000} \sum_{q \in \text{contours}(M)} \text{arcLength}(q) \quad [\text{m}].$$

Because L sums contour arc lengths, it is a boundary-length proxy, not a measured crack centreline length. Coverage ρ , panel area A_{panel} , and line density δ are

$$\rho = 100 \frac{|M|}{HW} \quad [\%], \quad A_{\text{panel}} = \frac{HWs^2}{10^6} \quad [\text{m}^2], \quad \delta = \frac{L}{A_{\text{panel}}} \quad [\text{m}/\text{m}^2].$$

Table 3 defines severity ℓ : select the row whose width interval contains $w_{\max}$, then use its deduction $d(\ell)$. A value on an interval boundary enters the higher severity level. The score is

$$\text{pCCS} = \max(0, 100 - d(\ell) - \min(25, 1.5\rho)).$$

It is a single-image, crack-only heuristic, not the ASTM D6433 Pavement Condition Index (PCI), which requires multi-distress sample-unit assessment.

Table 4

Ordered maintenance-support rules; the first match determines the output.

Condition	Action	Urgency	Planning window
$\ell \leq 2$ and $pCCS \geq 70$	Crack sealing	Routine	0–12 months
$\ell = 3$, or alligator with $\ell \leq 3$	Crack filling and patching	Moderate	0–6 months
$\ell = 4$, or $55 \leq pCCS < 70$	Mill and overlay	Urgent	0–3 months
$\ell \geq 5$, or $pCCS < 40$	Full-depth reclamation	Critical	Immediate
$pCCS < 55$	Mill and overlay	Urgent	0–3 months
Otherwise	Monitor; no immediate action	Routine	Next survey

The ordered rules in Table 4 generate an action, urgency, and planning window, with the first matching rule taking precedence over every later rule. First-match ordering is chosen over a scoring scheme because it yields a single traceable reason for each output. An engineer contesting a recommendation can be shown the one rule that fired and the values that satisfied it. A weighted combination would offer no such handle, since no individual quantity would explain the result. The rules are configurable prototype logic, not an agency-validated treatment-selection standard, and an agency adopting the framework would be expected to substitute its own table without altering anything upstream.

The JSON record retains the image identifier, timestamp, model phase, thresholds, scale, class probabilities, geometry, score, and maintenance output. Technical and executive reports use the same record. A language model may rephrase completed fields but cannot calculate or overwrite decision values. This protects the authoritative data path; it does not guarantee hallucination-free narration.

The record also assumes that the image it describes is what it claims to be. Provenance is only as trustworthy as the channel the imagery arrived through, and a tampered or spoofed input yields a perfectly traceable record of a wrong decision. This matters for aerial inspection in particular, where unmanned platforms introduce their own attack surface [27]. Acquisition-channel integrity is therefore a precondition of the present design rather than a property it provides. Generated text still requires consistency checks and engineer review. Event-level override logging and interface-level governance have not been fully evaluated.

4. Experiments and Results

This section reports task-level performance, a contextual comparison with the published reference, and a matched-threshold analysis that separates binarisation from the rest of the configuration. Classification behaviour and qualitative explanation cases follow. Every result below is subject to the source-overlap limitation established in Section 3.1, and it is restated wherever it changes how a number should be read.

4.1. Experimental setup and evaluation scope

Experiments used two NVIDIA Quadro RTX 4000 GPUs with 8 GB each (driver 595.71), PyTorch 2.12.0, and CUDA 13.0. Phase A used Adam with learning rate 3×10^{-4} , batch size 16, 150 epochs, and cosine decay. Phase B used AdamW with learning rate 10^{-4} , batch size 8, and 60 epochs. Phase A remained frozen, and the reported run used seed 42. Phase A contains 54.63 million parameters; the full stack contains 58.72 million, with 4.10 million trainable during Phase B. These are the reported rounded counts.

Reported task-decoupled inference latency is 37.67 ms per image: 12.12 ms for classification and detection at 224 pixels, 24.94 ms for CTAG-Seg at 384 pixels, and 0.61 ms for the deterministic decision layer. The decision layer therefore accounts for under 2% of inference time, so procedural auditability is close to free at this scale. These timings are measured on the hardware above and are not a validated field-deployment guarantee; they also exclude the optional language model.

Classification and detection use 1,500 typed test rows. Detection F1 uses intersection over union (IoU)

Table 5

Historical test-partition metrics. Panel (a) evaluates each phase at its own validation-selected threshold; panel (b) evaluates both at a common threshold, isolating binarisation from the rest of the configuration. Scores are percentages, differences are percentage points, and segmentation uses the 877 genuine-mask rows.

Metric	Phase A	Phase B	$\Delta(B-A)$
<i>(a) Selected operating points ($t_A^* = 0.75$, $t_B^* = 0.35$)</i>			
Classification macro F1	95.02	95.02	0.00
Detection F1	95.50	95.50	0.00
Segmentation Dice	82.17	83.40	+1.23
Segmentation IoU	72.02	73.81	+1.79
Segmentation precision	79.51	79.13	-0.38
Segmentation recall	87.22	90.62	+3.40
<i>(b) Matched threshold ($t = 0.50$)</i>			
Segmentation Dice	81.46	84.15	+2.69
Segmentation IoU	70.97	74.92	+3.95
Segmentation precision	75.79	83.23	+7.44
Segmentation recall	90.34	87.34	-3.00

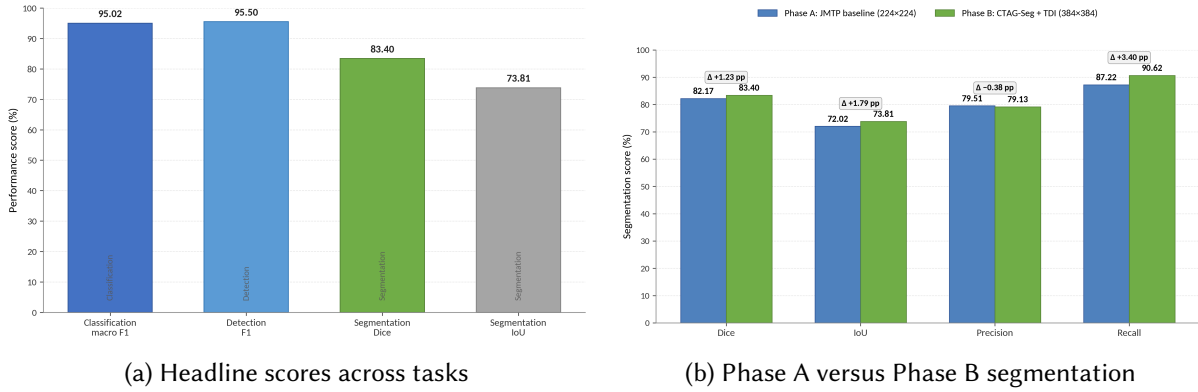


Figure 2: Historical test performance. (a) Classification macro F1 and detection F1 on 1,500 typed test rows, with segmentation Dice and IoU on 877 genuine-mask rows. (b) Segmentation at independently validation-selected thresholds ($t_A^* = 0.75$, $t_B^* = 0.35$); the direction of the precision-recall trade-off reverses at a matched threshold, as reported in panel (b) of Table 5. The source-overlap limitation of Section 3.1 applies throughout.

≥ 0.5 , confidence threshold 0.5, and confidence-ordered, one-to-one greedy matching. Segmentation uses the 877 genuinely mask-supervised test rows; per-image precision and recall are macro-averaged. The source-overlap limitation in Section 3.1 applies to every test result below.

4.2. Task-level performance

Table 5 compares complete operating configurations. Classification and detection are unchanged because Phase B does not update those branches. At independently validation-selected segmentation thresholds, Dice increases by 1.23 percentage points and IoU by 1.79 points; recall increases by 3.40 points while precision decreases by 0.38 points. The comparison combines architectural, training, resolution, and threshold effects and is not a component ablation.

A 1,000-resample percentile bootstrap gives 95% confidence intervals of 93.85–96.14% for classification macro F1 and, on mask-supervised rows, 82.39–84.45% for Dice, 72.71–75.05% for IoU, 78.08–80.39% for precision, and 89.68–91.44% for recall. These intervals describe resampling variability within the historical partition. They neither correct source overlap nor measure repeated-training variability; dependence between rows can further limit their interpretation.

Figure 2 summarises the headline scores across tasks in panel (a) and the segmentation comparison at the selected operating points in panel (b).

Table 6

Contextual comparison with Ashraf, Sophian, and Bawono [2]. Values are percentages and differences are percentage points; evaluation protocols are not controlled.

Metric	Published reference	TRACE-Pave	Difference
Classification F1 ^a	94.50	95.02	+0.52
Detection F1	94.40	95.50	+1.10
Segmentation Dice	94.70	83.40	−11.30
Segmentation IoU	92.30	73.81	−18.49
Segmentation precision	91.50	79.13	−12.37
Segmentation recall	90.70	90.62	−0.08

^aTRACE-Pave reports macro F1; the reference reports classification F1 without establishing an identical aggregation protocol. Differences are arithmetic only.

A radar summary of the same values is omitted: radar profiles compress per-class behaviour into a single contour, which is the weakness discussed in Section 4.5.

4.3. Contextual comparison with published results

Table 6 uses the final-model values in Tables 10 and 12 of Ashraf et al. [2]. Different splits, preprocessing, and metric aggregation make this a contextual comparison. Classification and detection differences cannot be attributed to CTAG-Seg or the decision layer. The published model reports substantially higher segmentation scores, by 11.30 points of Dice and 18.49 points of IoU. The gap is not attributed here to any single cause, but the two systems are not measured on comparable material: the segmentation metrics reported above are computed only on genuine-mask rows drawn from a single source, whereas the reference aggregates over its own corpus, and the training mixture used here includes untyped CRACK500 masks with a different annotation convention. Reporting the deficit is more useful here than explaining it away, and closing or accounting for it is part of the leakage-free retraining described in Section 5.

TRACE-Pave additionally implements a deterministic decision chain, structured JSON provenance, and classification Grad-CAM. These are functional characteristics rather than numerical performance gains, and they are therefore kept outside the metric comparison rather than being reported as additional rows in which TRACE-Pave scores and the reference does not. A capability present in one system and absent in the other is not a measurement, and presenting it as one would repeat the conflation this paper argues against.

4.4. Matched-threshold segmentation comparison

Both phases were evaluated on the threshold grid $\{0.30, 0.35, \dots, 0.85\}$. Panel (b) of Table 5 reports the matched point $t = 0.50$. Phase B has higher overlap and precision but lower recall, unlike the recall increase at independently selected operating points. This reversal shows why the threshold must accompany every segmentation result. Matching the threshold controls binarisation only; it does not isolate resolution, FiLM, ASPP, or individual losses.

4.5. Convergence and classification behaviour

Figure 3 shows composite training loss decreasing from approximately 0.58 to 0.21 over 60 epochs, with validation Dice reaching 83.52% at epoch 59. There is no sustained late-stage validation decline in this run, but a single trajectory does not establish optimisation stability across seeds.

Overall classification accuracy is 95.0%. Per-class recall is 92.29% for alligator, 94.81% for longitudinal, and 97.92% for transverse cracking. The largest confusion pathways are alligator to longitudinal (24 cases) and longitudinal to transverse (18 cases), as shown in Figure 4. Alligator has the lowest recall in this partition; this finding should not be treated as a source-independent class ranking.

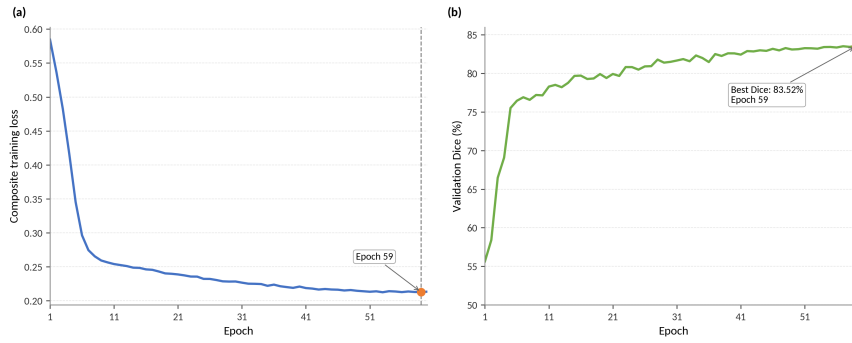


Figure 3: Reported CTAG-Seg training dynamics for the single-seed run: (a) composite training loss and (b) validation Dice over 60 epochs.

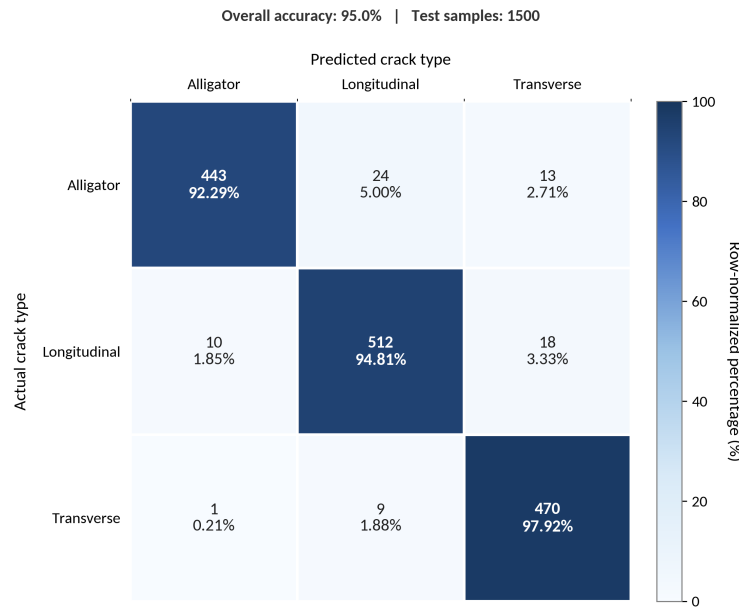


Figure 4: Classification confusion matrix on the 1,500-row typed test partition. Cells give counts with row-normalised percentages; rows are reference classes and columns are predictions.

Radar charts summarise tasks but can hide class-specific errors. Class maps show observation-level classification behaviour [28]; two-tier diagnostics organise classifier performance in two dimensions [29]; and cost curves examine sensitivity to error costs and class distributions [30]. These complementary methods are reserved for future evaluation.

4.6. Qualitative explanation analysis

Figure 5 contains exactly three independently normalised Grad-CAM examples. A longitudinal example is classified correctly at 99.68% confidence. The ground-truth transverse and alligator examples are misclassified at 98.98% and 93.50%, respectively. These failures show that a coherent-looking heatmap and high softmax confidence can accompany an incorrect prediction.

No batch-level localisation, deletion, insertion, or perturbation metric is reported, so the examples support failure inspection only, not quantitative claims about explanation quality, segmentation correctness, or calibrated human trust. Classification calibration has likewise not been measured. This matters more here than in a pure perception system. The decision layer consumes the class call without any confidence gate, so a confidently wrong crack type propagates into the severity rule and the maintenance lookup unchecked. Measuring calibration and adding such a gate are therefore coupled requirements rather than independent improvements.

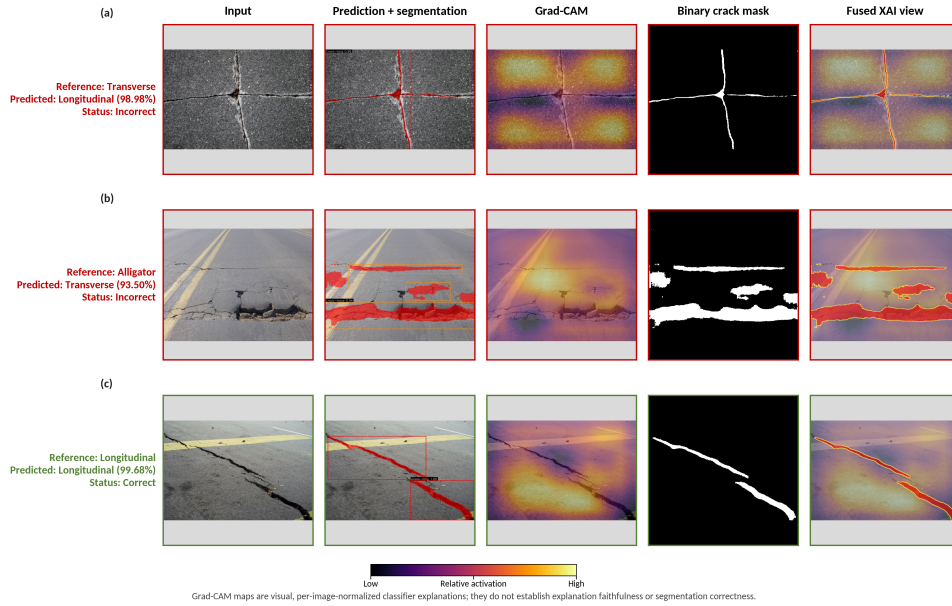


Figure 5: Qualitative classification diagnostics. Rows (a)–(b) are high-confidence errors; row (c) is correct.

5. Discussion

The main result is an explicit separation between prediction and decision authority. CTAG-Seg changes segmentation while retaining Phase A classification and detection outputs, and the reversal of the precision-recall trade-off between selected and matched thresholds confirms that this is a configuration-level result rather than an attribution of gains to any component. The practical consequence falls on the decision layer. False-positive mask areas alter coverage, width, severity, and the recommended treatment even when every downstream calculation is reproducible, so a screening bias in perception becomes a systematic bias in the recommendation. That propagation is visible precisely because the chain is explicit, which is the point of building it this way.

Procedural auditability makes the propagation inspectable but does not remove upstream error. An engineer can examine the mask, the scale, the geometric proxies, the severity interval, and the matched rule separately, and contest any one of them without discarding the rest. The supplied millimetres-per-pixel scale can nonetheless yield internally consistent yet physically wrong measurements, and contour length is not centreline length. The claim is therefore narrow and worth stating precisely: TRACE-Pave does not explain its neural computation better than prior systems, and its rules are simple by design rather than by achievement. What it offers is a structure in which each step of a maintenance decision has a recorded input, an identifiable rule, and a single place to be challenged.

5.1. Limitations

The strongest limitation is source overlap caused by splitting after oversampling. An audit of the pipeline found 975 source images shared between training and validation, 1,004 between training and test, and 458 between validation and test. The unique-source counts across the three partitions also exceed the number of distinct photographs available, which is itself sufficient proof of duplication. Because augmentation was applied online and only to training batches, test images were presented unmodified, but they depict roads and cracks the model had already seen. Classification and detection are the most affected, since every typed image was eligible for duplication, whereas segmentation is affected only through the single mask-supervised source. The per-class pattern is consistent with memorisation: transverse cracking was duplicated roughly three times more heavily than the other classes and also records the highest per-class recall. The per-class ranking is therefore not read as a statement about which distress type is intrinsically harder.

Leakage-free retraining with source-level partitions is the first priority, before any reported value

is interpreted as generalisation evidence. Retraining will shrink the evaluation set, since validation and test will hold each photograph once rather than in expanded form, and the headline figures are expected to fall. Four narrower limitations remain. Components of CTAG-Seg have not been ablated individually, so no single element can be credited for the refinement. Results come from one seed, and the reported intervals describe resampling within a fixed partition rather than training variability. Explanation quality rests on three qualitative cases with no localisation or perturbation metric, and classification calibration is unmeasured, which matters because the decision layer consumes the class call without a confidence gate. Finally, the pCCS and treatment rules are configurable prototypes rather than agency-validated standards, and no practitioner study has tested whether the authority boundary improves error detection or override quality in practice.

6. Conclusion

TRACE-Pave integrates established neural perception methods with deterministic, recorded condition-assessment rules. Its contribution lies in the auditable path from mask to measurement and recommendation, alongside an explicit engineer authority boundary. The reported configuration obtains 95.02% classification macro F1, 95.50% detection F1, and 83.40% segmentation Dice, but these results remain exploratory because the historical split contains source overlap. Task decoupling preserves classification and detection while changing segmentation behaviour; it does not establish individual component effects or explanation faithfulness. **Future work.** Four steps follow directly from the limitations above. Source-first retraining with repeated seeds must precede any generalisation claim. Controlled Phase B ablations and a full threshold sweep would attribute the refinement to specific components. Quantitative attribution metrics and a calibration study, paired with a confidence gate on the class call, would close the explanation gap. Practitioner evaluation would test whether the authority boundary improves error detection in practice, which is the claim this architecture ultimately rests on. Extension toward ASTM D6433-compliant multi-distress assessment requires engineering validation well beyond the present pCCS.

Data and Code Availability

The project repository is <https://github.com/tayyabrehman96/Explainable-pavement-Crack-Assessment>. Reproducing the historical results requires the corresponding checkpoint, split manifests, environment specification, and saved predictions; the corrected loader alone does not reproduce a leakage-free experiment.

Acknowledgments

This work was funded by “Bando a cascata Spoke 6, MOST-UniMore” (NextGeneration EU), within the Italian National PNRR, under grant IMPACT (CUP: E93C22001070001) and PNRR Next Generation EU, pursuant to Art. 8 of DM 630/2024, in collaboration with SPEE Srl of L’Aquila, Italy.

Declaration on Generative AI

During the preparation of this work, the authors used OpenAI tools for language editing, structural organisation, reference checking, figure presentation, and $\text{\LaTeX}$ formatting. The tools were not used to design the methodology, conduct experiments, or generate experimental data.

References

- [1] ASTM International, Standard practice for roads and parking lots pavement condition index surveys, Standard ASTM D6433-18, ASTM International, 2018. doi:10.1520/D6433-18.
- [2] A. Ashraf, A. Sophian, A. A. Bawono, Crack detection, classification, and segmentation on road pavement material using multi-scale feature aggregation and transformer-based attention mechanisms, *Construction Materials* 4 (2024) 655–675. doi:10.3390/constrmater4040036.
- [3] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, H. Ling, Feature pyramid and hierarchical boosting network for pavement crack detection, *IEEE Transactions on Intelligent Transportation Systems* 21 (2020) 1525–1535. doi:10.1109/TITS.2019.2910595.
- [4] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: Visual explanations from deep networks via gradient-based localization, in: *2017 IEEE International Conference on Computer Vision (ICCV)*, IEEE, 2017, pp. 618–626. doi:10.1109/ICCV.2017.74.
- [5] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, F. Herrera, Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, *Information Fusion* 58 (2020) 82–115. doi:10.1016/j.inffus.2019.12.012.
- [6] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, *Artificial Intelligence* 267 (2019) 1–38. doi:10.1016/j.artint.2018.07.007.
- [7] F. Doshi-Velez, B. Kim, Towards a rigorous science of interpretable machine learning, *arXiv preprint arXiv:1702.08608*, 2017. doi:10.48550/arXiv.1702.08608.
- [8] European Parliament and Council of the European Union, Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act), *Official Journal of the European Union*, L 2024/1689, 2024. URL: <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>.
- [9] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, P. Fung, Survey of hallucination in natural language generation, *ACM Computing Surveys* 55 (2023). doi:10.1145/3571730, article 248.
- [10] Y. Guo, Y. Xu, H. Cui, M. Dang, S. Li, Segment anything model-based crack segmentation using low-rank adaption fine-tuning, *Structural Health Monitoring* 24 (2025) 2579–2591.
- [11] N. Ma, Z. Song, Q. Hu, X. Tang, C. Zhang, R. Fan, L. Xie, WP-CrackNet: A collaborative adversarial learning framework for end-to-end weakly-supervised road crack detection, *Neurocomputing* 659 (2026). Article 131845.
- [12] S. Matarneh, F. Elghaish, A. Al-Ghraibah, E. Abdellatef, D. J. Edwards, An automatic image processing based on Hough transform algorithm for pavement crack detection and classification, *Smart and Sustainable Built Environment* 14 (2025) 1–22.
- [13] S. Huang, H. Chen, L. Yan, X. Zou, B. Li, Y. Bi, A review of the progress in machine vision-based crack detection and identification technology for asphalt pavements, *Digital Transportation and Safety* 4 (2025) 65–79.
- [14] E. Perez, F. Strub, H. de Vries, V. Dumoulin, A. Courville, FiLM: Visual reasoning with a general conditioning layer, *Proceedings of the AAAI Conference on Artificial Intelligence* 32 (2018) 3942–3951. doi:10.1609/aaai.v32i1.11671.
- [15] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40 (2018) 834–848. doi:10.1109/TPAMI.2017.2699184.
- [16] S. S. M. Salehi, D. Erdogmus, A. Gholipour, Tversky loss function for image segmentation using 3D fully convolutional deep networks, in: Q. Wang, Y. Shi, H.-I. Suk, K. Suzuki (Eds.), *Machine learning in medical imaging*, Springer, 2017, pp. 379–387. doi:10.1007/978-3-319-67389-9_44.
- [17] A. Chattopadhyay, A. Sarkar, P. Howlader, V. N. Balasubramanian, Grad-CAM++: Generalized gradient-based visual explanations for deep convolutional networks, in: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2018, pp. 839–847. doi:10.1109/

- [18] V. Petsiuk, A. Das, K. Saenko, RISE: Randomized input sampling for explanation of black-box models, in: *Proceedings of the British Machine Vision Conference 2018*, British Machine Vision Association, 2018, pp. 1–13. URL: <https://bmva-archive.org.uk/bmvc/2018/contents/papers/1064.pdf>.
- [19] S. Hooker, D. Erhan, P.-J. Kindermans, B. Kim, A benchmark for interpretability methods in deep neural networks, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché Buc, E. Fox, R. Garnett (Eds.), *Advances in neural information processing systems*, volume 32, Curran Associates, 2019, pp. 9737–9748. URL: https://papers.nips.cc/paper_files/paper/2019/hash/fe4b8556000d0f0cae99daa5c5c5a410-Abstract.html.
- [20] M. Gulfam, C. D. Seals, A. Vargis-Nordbeck, G. V. Dozier, Lightweight deep convolutional neural network for pavement crack recognition with explainability analysis, in: *Intelligent Systems, Blockchain, and Communication Technologies (ISBCom 2024)*, volume 1268 of *Lecture Notes in Networks and Systems*, Springer, 2025.
- [21] P. W. Koh, T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, P. Liang, Concept bottleneck models, in: *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020, pp. 5338–5348.
- [22] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, Y. Sekimoto, RDD2022: A multi-national image dataset for automatic road damage detection, *arXiv preprint arXiv:2209.08538*, 2022. doi:10.48550/arXiv.2209.08538.
- [23] A. Ashraf, A. Sophian, A. A. Shafie, T. S. Gunawan, N. N. Ismail, A. A. Bawono, RCD-IIUM: A comprehensive Malaysian road crack dataset for infrastructure analysis, in: *2024 9th International Conference on Mechatronics Engineering (ICOM)*, IEEE, 2024, pp. 200–206.
- [24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2017, pp. 936–944. doi:10.1109/CVPR.2017.106.
- [25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: *2017 IEEE International Conference on Computer Vision (ICCV)*, IEEE, 2017, pp. 2999–3007. doi:10.1109/ICCV.2017.324.
- [26] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: N. Navab, J. Hornegger, W. M. Wells, A. F. Frangi (Eds.), *Medical image computing and computer-assisted intervention—MICCAI 2015*, Springer, 2015, pp. 234–241. doi:10.1007/978-3-319-24574-4_28.
- [27] R. Butt, T. Rehman, N. Tariq, M. Humayun, Security of unmanned aerial vehicles, in: *Cybersecurity Issues and Challenges in the Drone Industry*, IGI Global, 2024. doi:10.4018/979-8-3693-0774-8.ch011.
- [28] J. Raymaekers, P. J. Rousseeuw, M. Hubert, Class maps for visualizing classification results, *Technometrics* 64 (2022) 151–165. doi:10.1080/00401706.2021.1927849.
- [29] G. Armano, A. Giuliani, A two-tiered 2D visual tool for assessing classifier performance, *Information Sciences* 463–464 (2018) 323–343. doi:10.1016/j.ins.2018.06.052.
- [30] C. Drummond, R. C. Holte, Cost curves: An improved method for visualizing classifier performance, *Machine Learning* 65 (2006) 95–130. doi:10.1007/s10994-006-8199-5.