CAN AN IMAGE BE ITS OWN REFERENCE? FREECRACK FOR LIGHTWEIGHT CRACK SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Thin cracks often resemble material joints, shadows, and surface texture, while foreground and background appearance shifts across images. Existing lightweight crack decoders aggregate multi-scale context but do not explicitly compare foreground and background evidence from the current input, leaving weak cracks fragmented and background textures falsely activated. We introduce FreeCrack, a compact decoder that treats each input as its own reference through an ordered evidence chain. Phase-preserving Lightweight Evidence Projection (PLEP) aligns coarse evidence to an unwarped shallow grid. In that shared geometry, Intra-image Task Prototype Mining (ITPM) constructs current-image crack and background prototypes and converts their similarity margin into dense evidence. Reliability-penalized Evidence Fusion (RPEF) then discounts dispersed token interactions before applying gated residual correction. The chain requires no support images, annotations beyond segmentation masks, or auxiliary supervision. Against the matched reproduced MixerCSeg baseline, FreeCrack raises four-benchmark average mIoU from 83.63% to 84.42% while reducing parameters by 42.5%. On TUT, mIoU rises from 81.89% to 82.68%. Fixed-width and targeted controls distinguish the effects of spatial alignment, current-image prototype source, residual correction, and reliability penalization. On DeepCrack, false positives and false negatives decrease by 11.1% and 4.8%, respectively. The decoder also improves RHACrackNet on DeepCrack and CrackMap and achieves higher Dice and IoU than matched SRF on DRIVE, spanning host architectures and a non-crack thin-structure domain. Together, these results support same-image foreground–background comparison as a compact decoder principle for resource-constrained thin-structure segmentation.

Keywords: Crack segmentation; Thin-structure segmentation; Same-image reference; Lightweight decoders; Prototype-based segmentation.

1 INTRODUCTION

Thin cracks can resemble material joints, shadows, or texture at a single pixel. Weak responses may be real cracks continuing across scales, whereas strong ones may arise from background texture. A compact segmenter must therefore establish cross-scale correspondence, identify foreground and background within the current image, and control uncertain evidence using its existing hierarchy.

Hierarchical models such as DeepCrack (Zou et al., 2019) and FPHBN (Yang et al., 2020) recover fine structures, while SCSegamba (Liu et al., 2025b) and MixerCSeg (Zhao et al., 2026) strengthen feature interactions. MixerCSeg combines TransMixer blocks, direction-guided edge convolutions, and spatial refinement fusion (SRF). SRF resizes coarse features and gates them with shallow attention, but does not form a foreground/background comparator for the current image. Such a comparator must avoid hard labels for ambiguous pixels and align coarse responses with the shallow reference; FreeCrack meets both requirements using soft structural evidence on a shared feature.

We formulate *same-image reference* as a compact decoder design for resolving foreground–background ambiguity from the current input. The methodological contribution is an ordered depen-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

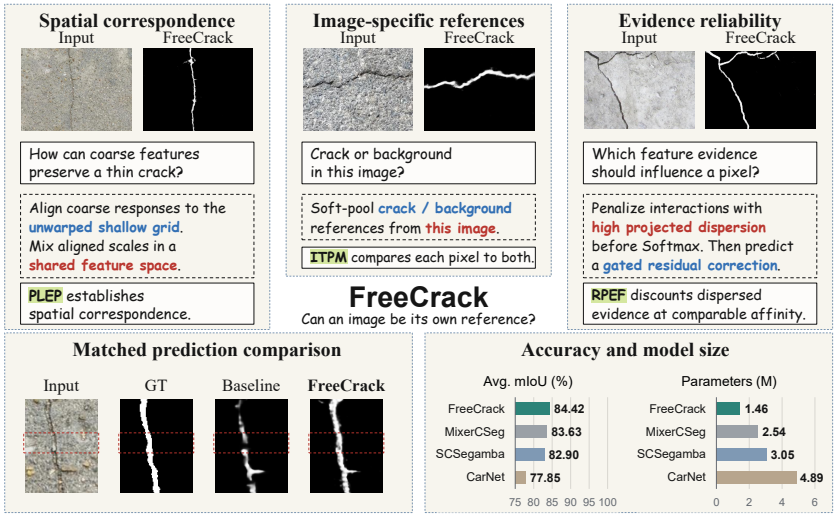

Figure 1: **FreeCrack overview.** PLEP aligns coarse evidence, ITPM builds current-image crack/background references, and RPEF discounts dispersed interactions. Lower panels show a matched prediction and the four-benchmark accuracy–size trade-off.

dency across three operations: coarse evidence is first aligned to an unwarped shallow coordinate frame, foreground and background prototypes are then constructed in that shared geometry, and the resulting evidence is finally corrected according to interaction reliability. This input-conditioned comparison requires no labeled support set, teacher, per-image class tag, or additional supervision.

Figure 1 maps the questions of alignment, reference construction, and interaction reliability to PLEP, ITPM, and RPEF. Figures 2 and 3 detail their computation and integration with the retained SRF path.

Our contributions are:

- We formulate **same-image reference** as input-conditioned foreground/background comparison for compact thin-structure decoders without labeled support, a teacher, or per-image class tags at inference.
- We realize it as a coordinated chain: PLEP establishes spatial correspondence, ITPM constructs image-conditioned prototypes, and RPEF applies residual correction with dispersion-based reliability modulation in one shallow-aligned feature space.
- We evaluate the design on five crack benchmarks with fixed-width and mechanism-specific controls, then test portability through RHACrackNet and DSCNet–DRIVE transfers. Average mIoU rises from 83.63% to 84.42% while parameters fall from 2.54M to 1.46M.

2 Related Work

Crack segmentation and compact dense prediction. Encoder–decoder (Long et al., 2015; Ronneberger et al., 2015; Badrinarayanan et al., 2017) and multi-scale models (Zhao et al., 2017; Chen et al., 2018; Wang et al., 2021) underpin dense prediction. Crack methods progress from structured forests (Shi et al., 2016) and convolutional detectors (Zhang et al., 2016) to hierarchical DeepCrack (Zou et al., 2019) and FPHBN (Yang et al., 2020); SCSegamba (Liu et al., 2025b) and MixerCSeg (Zhao et al., 2026) add selective and long-range interactions. Compact designs (Sandler et al., 2018; Ma et al., 2018; Howard et al., 2019; Han et al., 2020) motivate careful capacity allocation. FreeCrack concentrates new modeling in a shared eight-channel decoder with a compact encoder.

Spatial correspondence and feature selection. Attention provides channel, spatial, and long-range selection (Hu et al., 2018; Woo et al., 2018; Wang et al., 2018; Vaswani et al., 2017), including

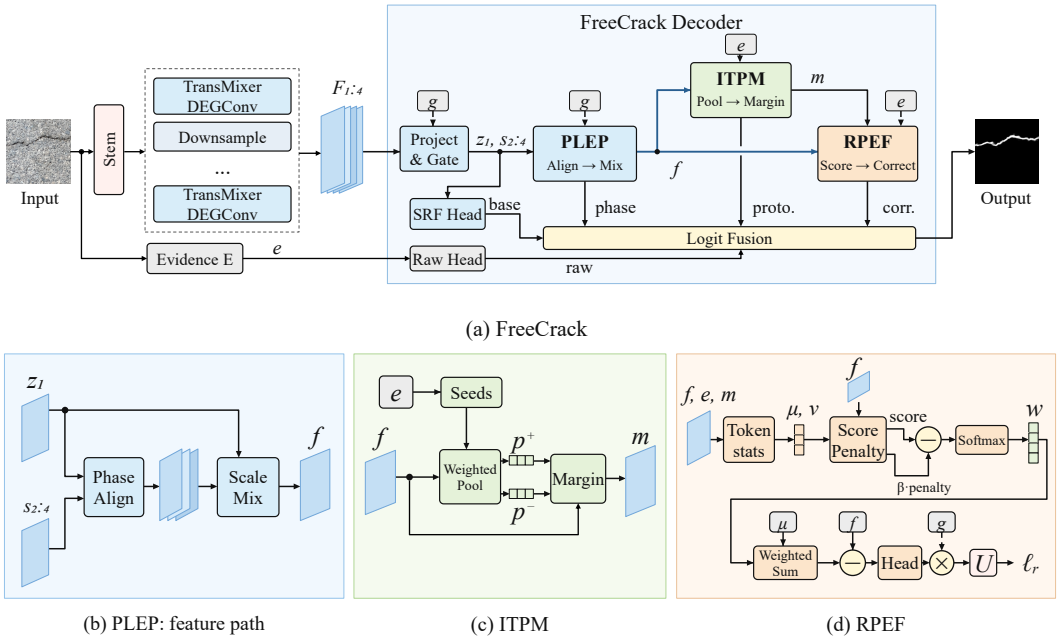

Figure 2: **FreeCrack architecture and evidence reuse.** (a) Gated coarse features feed retained SRF and PLEP; PLEP shares f with ITPM and RPEF, while ITPM supplies margin m. Five maps meet at Logit Fusion. (b) PLEP aligns coarse sources and mixes four scales. (c) ITPM forms foreground/background prototypes. (d) RPEF penalizes token scores before gated residual correction. Dashed arrows carry evidence or gates; e is shared evidence, g the effective gate, and U resizing. Repeated e/g ports share one source; line gaps denote crossings.

hierarchical dense-prediction formulations (Liu et al., 2021; Xie et al., 2021; Guo et al., 2022). Boundary models preserve local structure (Xie & Tu, 2015; Liu et al., 2017; Takikawa et al., 2019; Su et al., 2021), while fusion and guided upsampling connect alignment to high-frequency detail (Chen et al., 2024; Wimmer et al., 2026). PLEP uses the shallow feature for bounded alignment and per-pixel scale mixing, establishing the space shared by both later modules.

Prototypes and reliability. Prior prototype methods establish complementary forms of input adaptation. Self-Support Few-Shot Semantic Segmentation (Fan et al., 2022) refines support-derived prototypes with query predictions, while SIPE (Chen et al., 2022) mines image-specific prototypes from class-activation cues under image-level supervision. FreeCrack addresses compact fully supervised binary segmentation, where inference provides neither a labeled support image nor a per-image class tag. Its distinction lies in the conditional decoder chain: PLEP first maps coarse evidence to a fixed shallow geometry; ITPM then soft-pools crack and background prototypes from that shared feature using image-derived structural gates; and RPEF converts the prototype margin and complementary evidence into a dispersion-modulated residual. The prototype comparator is therefore coupled to spatial correspondence and reliability correction rather than deployed as an isolated prototype module.

3 FreeCrack

3.1 Shared Evidence and Prediction Paths

Let $x \in \mathbb{R}^{3\times H\times W}$ be an image. The encoder produces four feature maps $F_{1:4}$, each projected by an MLP and a convolution–group-normalization block into decoder features $z_s = \phi_s(F_s)$ with $d = 8$ channels. The reproduced MixerCSeg hierarchy uses widths $(16, 32, 64, 128)$; FreeCrack uses

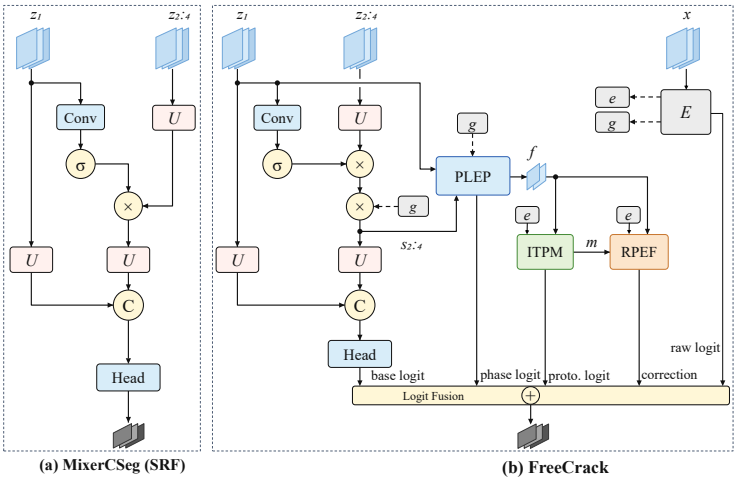

Figure 3: **From MixerCSeg SRF to FreeCrack.** (a) SRF gates coarse features with shallow attention. (b) FreeCrack conditions them with g, retains SRF prediction, and shares PLEP feature f with ITPM and RPEF; ITPM supplies margin m. Downward arrows are prediction maps, and g gates PLEP's phase prediction. C: concatenation; U: resizing; σ: Sigmoid; $\times$: multiplication.

$(16, 24, 48, 96)$, preserving the shallow width and concentrating evidence operations in the shared decoder space.

In the original SRF, shallow attention $a = \sigma(\mathrm{Conv}_{1\times1}(z_1))$ gates resized coarse features before prediction (Figure 3(a)). At the requested resolution, FreeCrack's evidence block E computes raw position/structure maps $r(x)$ (grayscale, local contrast, gradient, Laplacian, variance, and position), normalized structure-tensor coherence a_{coh}, and sine/cosine coordinate anchors $c(x)$. A small head predicts support, background, and concordance gates:

$$[g_{\mathrm{sup}}, g_{\mathrm{bg}}, g_{\mathrm{con}}] = \sigma(h_g([r(x), a_{\mathrm{coh}}, c(x)])), \qquad g = g_{\mathrm{sup}}(1 - g_{\mathrm{bg}}) \left[(1-\eta) + \eta g_{\mathrm{con}} \frac{1 + a_{\mathrm{coh}}}{2}\right]. \tag{1}$$

The evidence bundle e in the figures contains these maps and derived gates. With $U_{s\to1}$ resizing to the shallow decoder grid, the coarse sources become

$$s_k = U_{k\to1}(z_k) \odot a \odot g, \qquad k \in \{2, 3, 4\}. \tag{2}$$

The gated sources retain an SRF prediction path and feed PLEP for shared evidence (Figures 3(b) and 2(a)). Throughout, U denotes bilinear resizing to the receiving branch's resolution.

3.2 PLEP: Phase-Preserving Lightweight Evidence Projection

Thin structures provide useful cross-scale evidence when their responses occupy corresponding positions. PLEP therefore establishes correspondence before prototype mining. *Phase-preserving* retains shallow positions as an unwarped reference and applies bounded two-dimensional offsets only to each coarse source s_k:

$$\Delta_k = \delta_{\max} \tanh h_{\Delta,k}([z_1, s_k, |z_1 - s_k|, G(z_1), G(s_k)]), \qquad \bar{s}_k = \mathrm{grid_sample}(s_k, \mathcal{G} + \Delta_k). \tag{3}$$

Here G denotes normalized Sobel magnitude, $\mathcal{G}$ is the sampling grid, and offsets are converted from pixel units to grid coordinates. Each offset component is bounded by $\delta_{\max}$. The three aligned sources and the unwarped reference enter Scale Mix (Figure 2(b)):

$$\omega = \mathrm{softmax}_{\mathrm{scale}}\, h_\omega([z_1, \bar{s}_2, \bar{s}_3, \bar{s}_4, G(z_1), G(\bar{s}_2), G(\bar{s}_3), G(\bar{s}_4)]), \qquad f = \omega_1 \odot z_1 + \sum_{k=2}^{4} \omega_k \odot \bar{s}_k. \tag{4}$$

The z_1 bypass preserves shallow detail while aligned sources provide context; f serves both downstream modules. PLEP also supplies a phase prediction

$$\ell_\phi = U(g) \odot h_\phi([U(z_1), U(\bar{s}_2), U(\bar{s}_3), U(\bar{s}_4), U(f)]). \tag{5}$$

Alignment and scale scoring thus remain in the shared d-channel space.

3.3 ITPM: Intra-Image Task Prototype Mining

ITPM turns f into an image-specific foreground/background comparator. With no labeled support at inference, a hard preliminary mask could make ambiguous pixels define their own reference. The shared structural evidence instead supplies continuous support and background seeds

$$s^+ = \mathrm{clip}\left(g\frac{1+a_{\mathrm{coh}}}{2}, 0, 1\right), \qquad s^- = \mathrm{clip}(g_{\mathrm{bg}} + (1-g_{\mathrm{sup}})(1-a_{\mathrm{coh}}), 0, 1)\,. \tag{6}$$

The seeds weight all positions to form two normalized prototypes (Figure 2(c)):

$$\pi_i^\pm = \frac{\exp(\tau s_i^\pm)}{\sum_j \exp(\tau s_j^\pm)}, \qquad p^\pm = \mathrm{norm}\left(\sum_i \pi_i^\pm \,\mathrm{norm}(f_i)\right). \tag{7}$$

The direct f input to Margin compares each position with both prototypes:

$$m_i = \cos(f_i, p^+) - \cos(f_i, p^-), \qquad \ell_p = U(m). \tag{8}$$

The image-specific margin contributes a dense prediction and supplies prototype-confidence seed $\sigma(m)$ to RPEF.

3.4 RPEF: Reliability-Penalized Evidence Fusion

RPEF is a gated residual corrector with token selection modulated by query-projected dispersion, a consistency proxy for each query's evidence. The residual path supplies local correction, whereas dispersion controls which evidence influences each query. From f, the evidence bundle, and ITPM margin, it constructs four seed maps:

$$q_1 = g, \quad q_2 = g_{\mathrm{sup}}\frac{1+a_{\mathrm{coh}}}{2}, \quad q_3 = \mathrm{mean}_{\mathrm{channel}}\, c(x), \quad q_4 = \sigma(m). \tag{9}$$

All seeds are clipped to $[0, 1]$. For each seed j, Token Stats computes a normalized prototype and its channelwise dispersion:

$$\begin{aligned} \alpha_{j,i} &= \mathrm{softmax}_i(\tau q_{j,i}), \qquad \mu_j = \mathrm{norm}\left(\sum_i \alpha_{j,i}\,\mathrm{norm}(f_i)\right), \\ v_j &= \max\left(\sum_i \alpha_{j,i}(\mathrm{norm}(f_i) - \mu_j)^2, \epsilon\right). \end{aligned} \tag{10}$$

The projected query $u_i = \mathrm{norm}(W_q f_i)$ scores each token. The first subtraction in Figure 2(d) applies the reliability penalty before Softmax:

$$b_{j,i} = \frac{u_i^\top \mu_j}{\sqrt{d}} - \beta\sqrt{\frac{\max((u_i^2)^\top v_j, \epsilon)}{d}}, \qquad w_{j,i} = \mathrm{softmax}_j(b_{j,i}), \qquad t_i = \sum_{j=1}^{4} w_{j,i}\mu_j. \tag{11}$$

High dispersion reduces a token's influence in the corresponding query direction; the aggregate t yields a residual relative to f:

$$\ell_r = U(g \odot h_r(t - f))\,. \tag{12}$$

The effective gate controls where correction acts, and the learned head maps the feature difference to one logit channel. Cached seeds and gate use stop-gradient; f and token statistics remain differentiable. We set $\epsilon = 10^{-6}$ for numerical stability.

Table 1: Four-benchmark comparison. Reproduced MixerCSeg and FreeCrack form the matched-protocol pair; other methods provide context. mIoU is primary, with ODS, OIS, and F1 as complementary summaries. Bold marks each column maximum.

Method	DeepCrack				CamCrack789				CrackMap				Crack500			
	mIoU	ODS	OIS	F1	mIoU	ODS	OIS	F1	mIoU	ODS	OIS	F1	mIoU	ODS	OIS	F1
CarNet	0.888	0.869	0.870	0.894	0.826	0.783	0.785	0.725	0.757	0.684	0.712	0.640	0.643	0.549	0.592	0.364
RINDNet	0.839	0.800	0.827	0.838	0.814	0.763	0.773	0.605	0.743	0.675	0.694	0.670	0.738	0.647	0.648	0.712
MambaVision	0.899	0.885	0.905	0.880	0.815	0.760	0.763	0.708	0.774	0.716	0.766	0.686	0.702	0.630	0.648	0.460
SCSegamba	0.902	0.894	0.899	0.911	0.827	0.792	0.794	0.788	0.809	0.774	0.777	0.768	0.778	0.724	0.737	0.755
MixerCSeg	0.915	0.909	0.920	0.921	0.841	0.812	0.820	0.824	0.812	0.778	0.781	0.782	0.782	0.728	0.748	**0.776**
MixerCSeg (reproduced)	0.914	0.908	0.920	0.922	0.839	0.809	0.833	0.826	0.810	0.774	0.780	0.777	0.782	0.728	0.762	**0.776**
FreeCrack (Ours)	**0.923**	**0.918**	**0.932**	**0.929**	**0.848**	**0.821**	**0.850**	**0.835**	**0.823**	**0.793**	**0.804**	**0.796**	**0.783**	**0.729**	**0.763**	0.775

3.5 Five-Branch Logit Fusion

The retained SRF and raw-image evidence branches complement the three evidence modules:

$$\ell_{\text{base}} = h_{\text{base}}([U(z_1), U(s_2), U(s_3), U(s_4)]), \qquad \ell_{\text{raw}} = h_{\text{raw}}(e_{\text{out}}). \tag{13}$$

Here e_{out} contains output-resolution evidence maps. Logit Fusion combines five spatially matched, single-channel maps:

$$\hat{\ell} = \ell_{\text{base}} + \lambda_\phi \ell_\phi + \lambda_{\text{raw}} \ell_{\text{raw}} + \lambda_p \ell_p + \lambda_r \ell_r. \tag{14}$$

The four λ values are learned gains. FreeCrack retains the BCE-with-logits and Dice objective; supervision through the final prediction requires neither extra annotations nor auxiliary losses.

4 Experimental Setup

Datasets and protocol. We evaluate on DeepCrack (Liu et al., 2019), CamCrack789 (Zhu et al., 2024), CrackMap (Katsamenis et al., 2023), Crack500 (Zhang et al., 2016), and TUT (Liu et al., 2025a). The network was trained for 50 epochs, and the checkpoint achieving the best performance on the validation set was selected for final testing. CPU execution is restricted to a single thread and each training run uses at most one visible GPU.

Metrics. We use mean Intersection-over-Union (mIoU) as the primary segmentation endpoint throughout the main comparison and all component and control studies. ODS, OIS, F1, precision, and recall provide complementary threshold-based views of the predictions. Parameter count, FPS, and latency characterize the accuracy–efficiency trade-off. The main comparison therefore evaluates whether FreeCrack improves mIoU while operating under a smaller parameter budget.

Implementation. FreeCrack uses stage widths $(16, 24, 48, 96)$ and an eight-channel decoder. PLEP offsets are bounded by 1.5 pixels; $\tau = 8$, $\eta = 0.35$, and $\beta = 1.5$. The raw, prototype, and reliability gains are initialized to 0.1, 0.1, and 0.05 and optimized with the network.

5 Main Results

Accuracy under a smaller parameter budget. The matched comparison raises four-benchmark average mIoU from 0.8363 to 0.8442 while reducing parameters from 2.54M to 1.46M (42.5%). Gains are $+0.0088$ on DeepCrack, $+0.0086$ on CamCrack789, $+0.0131$ on CrackMap, and $+0.0013$ on Crack500 (Table 1); the largest occurs on textured CrackMap, where a current-image foreground/background reference is especially relevant. Full component results appear in Supplementary Table S1.

5.1 Qualitative Comparison

Figure 4 compares recorded predictions from CarNet (Li et al., 2024), RINDNet (Pu et al., 2021), MambaVision (Hatamizadeh & Kautz, 2025), SCSegamba, reproduced MixerCSeg, and FreeCrack under each method's checkpoint and pipeline. FreeCrack completes the highlighted bend in row 1, keeps CrackMap responses on annotated trajectories with fewer activations near aggregate boundaries in rows 2–4, and retains the DeepCrack trunk and curved trace in rows 5–6.

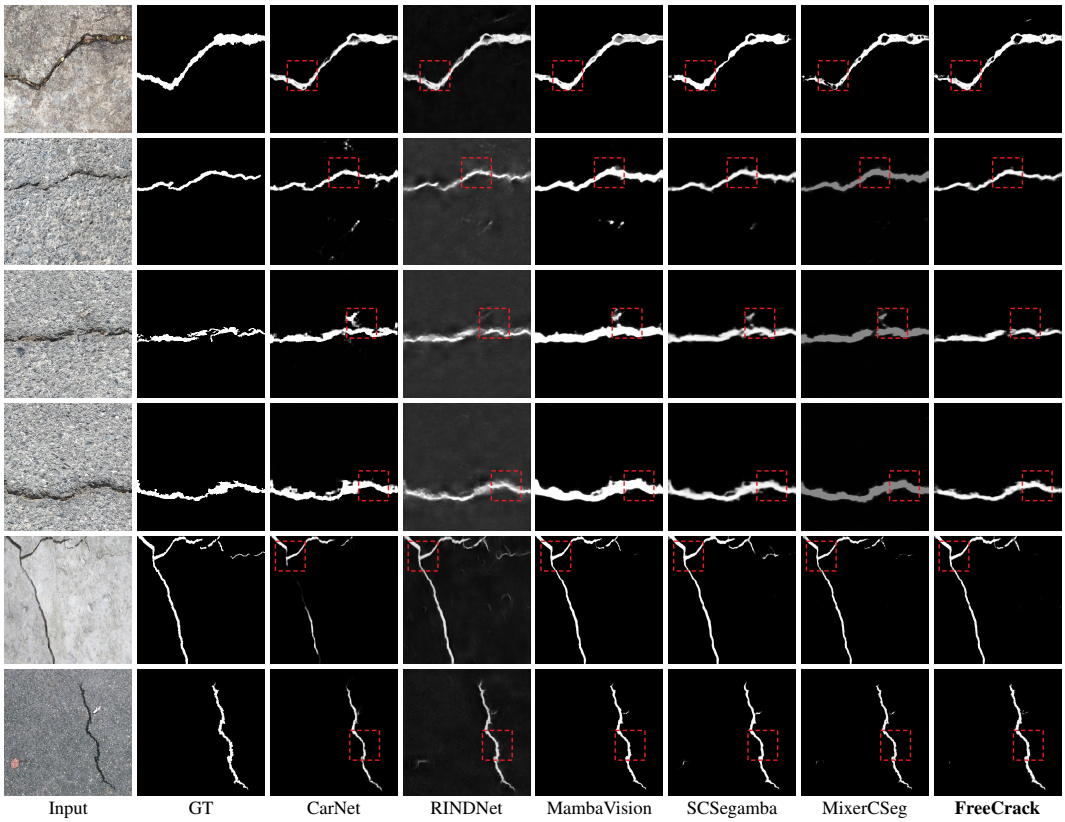

Figure 4: **Qualitative crack segmentation on selected test images.** The rows show CamCrack789 (row 1), CrackMap (rows 2–4), and DeepCrack (rows 5–6). Red boxes identify matching regions across predictions. Probability maps use a fixed grayscale range, and complete image frames are resized to common display cells. FreeCrack concentrates its response along the annotated crack through bends and junctions while limiting off-crack texture responses.

5.2 Additional-Dataset Evaluation

Evaluation on an additional benchmark. On TUT, FreeCrack improves mIoU from 0.8189 to 0.8268 and F1 from 0.8014 to 0.8119; precision and recall also rise (Table 2).

Table 2: Additional-dataset evaluation on TUT.

Method	mIoU	ODS	OIS	F1	Prec.	Recall
MixerCSeg	0.8189	0.7778	0.7929	0.8014	0.7853	0.8181
FreeCrack	**0.8268**	**0.7889**	**0.8050**	**0.8119**	**0.7997**	**0.8245**
Gain	+0.0079	+0.0111	+0.0121	+0.0106	+0.0144	+0.0065

6 Mechanism Analysis

6.1 Component Configurations

Table 3 compares configurations, size, and inference cost. Across four benchmarks, the complete chain has the highest evaluated average mIoU, 0.8442, versus 0.8392 for PLEP, 0.8416 for PLEP+ITPM, and 0.8411 for PLEP+RPEF (Supplementary Table S1). ITPM is geometry-conditioned: its change reverses from $\Delta_{\mathrm{I}|\emptyset} = -0.0024$ without PLEP to $\Delta_{\mathrm{I}|\mathrm{P}} = +0.0024$ after PLEP aligns pooled features and queried locations on one grid. Because PLEP+ITPM also uses the

Table 3: Component configurations on DeepCrack and CamCrack789. Checkmarks mark active evidence components; size, throughput, and latency are reported for each configuration. The full configuration is FreeCrack.

PLEP	ITPM	RPEF	Params (M)	FPS	Lat. (ms)	DeepCrack				CamCrack789			
						mIoU	ODS	OIS	F1	mIoU	ODS	OIS	F1
			2.54	8.07	123.89	0.9138	0.908	0.920	0.922	0.8395	0.809	0.833	0.826
✓			2.54	10.30	97.12	0.9156	0.910	0.924	0.924	0.8444	0.817	0.841	0.830
	✓		2.54	**13.65**	**73.28**	0.9101	0.904	0.918	0.915	0.8369	0.806	0.830	0.824
		✓	2.52	4.03	248.40	0.9144	0.909	0.921	0.922	0.8382	0.807	0.833	0.827
✓	✓		**1.36**	12.52	79.86	0.9196	0.915	0.929	0.927	0.8465	0.819	0.847	0.831
✓		✓	2.51	3.75	266.95	0.9196	0.915	0.929	0.926	0.8458	0.818	0.845	0.833
	✓	✓	2.52	7.93	126.10	0.9130	0.907	0.920	0.921	0.8387	0.808	0.831	0.828
✓	✓	✓	1.46	11.11	90.00	**0.9226**	**0.918**	**0.932**	**0.929**	**0.8481**	**0.821**	**0.850**	**0.835**

narrow allocation, this is a descriptive conditional contrast; the zero-offset control is the matched alignment intervention.

6.2 Controlled Evidence for the Full Chain

With encoder width fixed, the complete decoder improves all four benchmarks over original SRF, raising average mIoU from 0.8353 to 0.8442. Current-image prototypes add 0.0020 over learned image-independent prototypes on average and 0.0073 on CrackMap; learned offsets add 0.0024. RPEF's residual adds 0.0034, with dispersion penalization adding 0.0011. Base $\rightarrow$ PLEP $\rightarrow$ PLEP+ITPM $\rightarrow$ Full yields $+0.0029$, $+0.0024$, and $+0.0026$, while PLEP–ITPM gives the largest descriptive configuration contrast, 0.0048. Because PLEP+ITPM also uses the narrow allocation, this contrast remains descriptive. The full model leads all fixed-width and single-intervention controls with 1.08M fewer parameters than the reproduced baseline (Table 3; Supplementary Tables S1 and S4).

6.3 Error-Type Evidence

On DeepCrack, false positives fall from 85,439 to 75,960 and false negatives from 76,495 to 72,825; false positives also fall from 207,328 to 178,205 on CamCrack789 and from 56,736 to 47,152 on CrackMap. DeepCrack's joint reduction shows improved coverage alongside fewer false detections, consistent with prototype comparison and gated evidence correction.

7 Additional Evidence

7.1 Perturbation Robustness

FreeCrack retains positive mIoU margins on all four datasets under Gaussian noise, blur, contrast reduction, and JPEG compression; CrackMap shows especially large margins under reduced brightness and motion blur (Table 4). Corruptions are applied only to test images, while the ground-truth masks remain unchanged. On the 8-bit RGB scale, Gaussian noise uses $\sigma = 10$ with random seed 123; Gaussian blur uses a 5×5 kernel; motion blur uses a horizontal 9×9 averaging kernel; brightness is multiplied by 0.75; contrast is multiplied by 0.70 around intensity 127.5; and JPEG compression uses quality 40. Corrupted values are clipped to $[0, 255]$ before inference.

7.2 Additional Fixed-Seed Results

Across seeds 42, 2026, and 2027, mean paired gains remain positive on DeepCrack, CamCrack789, and Crack500: 0.0086 ± 0.0010, 0.0085 ± 0.0020, and 0.0009 ± 0.0008, respectively (sample standard deviation over seeds; Supplementary Table S3).

Table 4: Perturbation robustness. Values are mIoU gains of FreeCrack over MixerCSeg under matched test-image corruptions: Gaussian noise with $\sigma = 10$, 5×5 Gaussian blur, horizontal 9×9 motion blur, brightness factor 0.75, contrast factor 0.70, and JPEG quality 40.

Perturbation	DeepCrack	CamCrack789	CrackMap	Crack500
Gaussian noise	**+0.0097**	**+0.0080**	**+0.0106**	**+0.0019**
Gaussian blur	**+0.0054**	**+0.0053**	**+0.0212**	**+0.0002**
Motion blur	-0.0049	**+0.0053**	**+0.0271**	0.0000
Brightness down	**+0.0086**	**+0.0081**	**+0.0743**	0.0000
Contrast down	**+0.0101**	**+0.0101**	**+0.0169**	**+0.0032**
JPEG40	**+0.0086**	**+0.0074**	**+0.0140**	**+0.0015**

7.3 Plug-in Transfer

On RHACrackNet (Zhu et al., 2024), the plug-in raises mIoU from 0.9741 to 0.9769 on DeepCrack and from 0.9409 to 0.9435 on CrackMap as size changes from 1.665M to 1.666M (Table 5). On DRIVE Staal et al. (2004), 1.132M-parameter DSCNet+FreeCrack Qi et al. (2023) reaches 0.8093 Dice and 0.6799 IoU, above compact SRF at 0.8078 and 0.6779 while retaining 99.37% of native DSCNet Dice with 45.96% fewer parameters (Supplementary Table S5). These transfers span both host and image-domain changes.

Table 5: Compatible-decoder transfer on RHACrackNet. Baseline and plug-in results use the matched RHACrackNet setting.

Dataset	Variant	mIoU	Params (M)	FPS
DeepCrack	RHACrackNet	0.9741	1.665	119.20
DeepCrack	+FreeCrack	**0.9769**	1.666	114.23
CrackMap	RHACrackNet	0.9409	1.665	117.39
CrackMap	+FreeCrack	**0.9435**	1.666	113.93

8 Discussion

Replacing current-image prototypes with image-independent learned prototypes reduces average mIoU by 0.0020 and by 0.0073 on textured CrackMap. The larger CrackMap margin matches the input-dependent ambiguity targeted by same-image comparison. PLEP supplies the shared coordinate frame, explaining why ITPM is most effective after alignment.

Removing the RPEF residual path reduces average mIoU by 0.0034, while removing dispersion penalization accounts for a further 0.0011 refinement. Residual correction supplies the main update and dispersion controls its local influence. With the fixed-width narrow-SRF control, these interventions support an ordered chain beyond encoder narrowing or any isolated component.

9 Conclusion

FreeCrack compares current-image crack and background references in a shallow-aligned feature space. Its PLEP–ITPM–RPEF chain raises four-benchmark average mIoU from 0.8363 to 0.8442 while reducing parameters from 2.54M to 1.46M. Controls trace the gain to alignment, prototype source, residual correction, and reliability; TUT, RHACrackNet, and compact DSCNet–DRIVE results extend the evidence across benchmarks, hosts, and a non-crack thin-structure domain.

AI USE STATEMENT

We declare that no artificial intelligence (AI) tools, including generative AI or large language models, were used in the writing, editing, or preparation of this manuscript.

ETHICS STATEMENT

This work uses public benchmarks, recruits no participants, collects no private data, and releases no new dataset. Its primary application is structural visual inspection under domain-appropriate engineering review.

REPRODUCIBILITY STATEMENT

The Supplementary Material provides full component results, additional fixed-seed comparisons, targeted mechanism controls, descriptive configuration and parameter analyses, and cross-domain decoder transfer on DRIVE.

REFERENCES

Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision*, pp. 801–818, 2018.

Linwei Chen, Ying Fu, Lin Gu, Chenggang Yan, Tatsuya Harada, and Gao Huang. Frequency-aware feature fusion for dense image prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10763–10780, 2024. doi: 10.1109/TPAMI.2024.3449959.

Qi Chen, Lingxiao Yang, Jian-Huang Lai, and Xiaohua Xie. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4288–4298, 2022.

Qi Fan, Wenjie Pei, Yu-Wing Tai, and Chi-Keung Tang. Self-support few-shot semantic segmentation. In *Computer Vision – ECCV 2022*, volume 13679 of *Lecture Notes in Computer Science*, pp. 701–719. Springer, 2022. doi: 10.1007/978-3-031-19800-7_41.

Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zheng-Ning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. In *Advances in Neural Information Processing Systems*, 2022.

Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1580–1589, 2020.

Ali Hatamizadeh and Jan Kautz. MambaVision: A hybrid Mamba-Transformer vision backbone. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25261–25270, 2025.

Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, and Hartwig Adam. Searching for mobilenetv3. In *IEEE/CVF International Conference on Computer Vision*, pp. 1314–1324, 2019.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.

Iason Katsamenis, Eftychios Protopapadakis, Nikolaos Bakalos, Andreas Varvarigos, Anastasios Doulamis, Nikolaos Doulamis, and Athanasios Voulodimos. A few-shot attention recurrent residual u-net for crack segmentation. In *Advances in Visual Computing*, volume 14361 of *Lecture Notes in Computer Science*, pp. 199–209. Springer, 2023. doi: 10.1007/978-3-031-47969-4_16.

Kai Li, Jie Yang, Siwei Ma, Bo Wang, Shanshe Wang, Yingjie Tian, and Zhiquan Qi. Rethinking lightweight convolutional neural networks for efficient and high-quality pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems*, 25(1):237–250, 2024. doi: 10.1109/TITS.2023.3307286.

Hui Liu, Chen Jia, Xu Cheng, Xiufeng Liu, and Fan Shi. Serial local patterns and irregular dependencies extract and cascaded fusion network for structural crack segmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2025a. doi: 10.1109/ICASSP49660.2025.10887799.

Hui Liu, Chen Jia, Fan Shi, Xu Cheng, and Shengyong Chen. Scsegamba: Lightweight structure-aware vision mamba for crack segmentation in structures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025b.

Yahui Liu, Jian Yao, Xiaohu Lu, Renping Xie, and Li Li. Deepcrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing*, 338:139–153, 2019. doi: 10.1016/j.neucom.2019.01.036.

Yun Liu, Ming-Ming Cheng, Xiaowei Hu, Kai Wang, and Xiang Bai. Richer convolutional features for edge detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3000–3009, 2017.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, 2015.

Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *European Conference on Computer Vision*, pp. 116–131, 2018.

Mengyang Pu, Yaping Huang, Qingji Guan, and Haibin Ling. RINDNet: Edge detection for discontinuity in reflectance, illumination, normal and depth. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6879–6888, 2021.

Yaolei Qi, Yuting He, Xiaoming Qi, Yuan Zhang, and Guanyu Yang. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6070–6079, 2023.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

Yong Shi, Limeng Cui, Zhiquan Qi, Fan Meng, and Zhensong Chen. Automatic road crack detection using random structured forests. *IEEE Transactions on Intelligent Transportation Systems*, 17 (12):3434–3445, 2016.

Joes Staal, Michael D. Abràmoff, Meindert Niemeijer, Max A. Viergever, and Bram van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 2004. doi: 10.1109/TMI.2004.825627.

Zhuo Su, Wei Liu, Zitong Yu, Dewen Hu, Qing Liao, Qi Tian, Matti Pietikäinen, and Li Liu. Pixel difference networks for efficient edge detection. In *IEEE/CVF International Conference on Computer Vision*, pp. 5117–5127, 2021.

Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *IEEE/CVF International Conference on Computer Vision*, pp. 5229–5238, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3349–3364, 2021.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2018.

Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. Anyup: Universal feature upsampling. In *International Conference on Learning Representations*, 2026.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *European Conference on Computer Vision*, pp. 3–19, 2018.

Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems*, 2021.

Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *IEEE International Conference on Computer Vision*, pp. 1395–1403, 2015.

Fan Yang, Lei Zhang, Sijia Yu, Danil Prokhorov, Xue Mei, and Haibin Ling. Feature pyramid and hierarchical boosting network for pavement crack detection. *IEEE Transactions on Intelligent Transportation Systems*, 21(4):1525–1535, 2020.

Lei Zhang, Fan Yang, Yimin Daniel Zhang, and Ying Julie Zhu. Road crack detection using deep convolutional neural network. In *IEEE International Conference on Image Processing*, pp. 3708–3712, 2016.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2881–2890, 2017.

Zilong Zhao, Zhengming Ding, Pei Niu, Wenhao Sun, and Feng Guo. Mixercseg: An efficient mixer architecture for crack segmentation via decoupled mamba attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Guijie Zhu, Jiacheng Liu, Zhun Fan, Duan Yuan, Peili Ma, Meihua Wang, Weihua Sheng, and Kelvin C. P. Wang. A lightweight encoder–decoder network for automatic pavement crack detection. *Computer-Aided Civil and Infrastructure Engineering*, 39(12):1743–1765, 2024. doi: 10.1111/mice.13103.

Qin Zou, Zheng Zhang, Qingquan Li, Xianbiao Qi, Qian Wang, and Song Wang. Deepcrack: Learning hierarchical convolutional features for crack detection. *IEEE Transactions on Image Processing*, 28(3):1498–1512, 2019.

SUPPLEMENTARY MATERIAL FOR CAN AN IMAGE BE ITS OWN REFERENCE? FREECRACK FOR LIGHTWEIGHT CRACK SEGMENTATION

Anonymous authors
Paper under double-blind review

1 REPRODUCIBLE EVALUATION PROTOCOL

Data and preprocessing. All crack experiments use fixed train, validation, and test splits for DeepCrack, CamCrack789, CrackMap, Crack500, and TUT. Inputs and binary masks are resized to 512×512. Within every matched comparison, the dataset lists, loader, augmentations, normalization, and evaluator are unchanged; no test-time adaptation is used.

Training and checkpointing. Each model is trained for 50 epochs with batch size 1 using AdamW (initial learning rate 5×10^{-4}, weight decay 10^{-2}) and PolyLR with minimum learning rate 10^{-6}. The objective is $0.87\,\mathcal{L}_{\mathrm{BCE}} + 0.13\,\mathcal{L}_{\mathrm{Dice}}$. Main component and control studies use seed 42; the additional fixed-seed study uses seeds 2026 and 2027 under the same schedule. The checkpoint with the highest validation mIoU is retained for final test evaluation. CPU execution is restricted to one thread and each run uses one visible GPU.

Metric aggregation. The evaluator sweeps thresholds from 0.00 to 0.99 in increments of 0.01. At each threshold, foreground and background IoU are averaged per image and then across images; mIoU is the maximum of this curve. ODS is the maximum dataset-level F1 over the same thresholds, OIS is the mean of the per-image maximum F1 values, and the reported F1, precision, and recall are computed from pooled confusion counts at the ODS threshold. Parameter count is $\sum_p \mathrm{numel}(p)$ over all model parameters. All paired latency and FPS measurements use the same image size, batch size, warm-up, and timing loop.

2 FULL COMPONENT RESULTS

Table S1 gives the parameter count and four-benchmark mIoU of all eight evaluated component configurations. The complete evidence chain has the highest average among these configurations.

Table S1: Parameter count and four-benchmark mIoU of the evaluated component configurations. The full evidence chain achieves the highest average.

PLEP	ITPM	RPEF	Params (M)	DeepCrack	CamCrack789	CrackMap	Crack500	Avg. mIoU
			2.54	0.9138	0.8395	0.8096	0.7822	0.8363
✓			2.54	0.9156	0.8444	0.8116	**0.7852**	0.8392
	✓		2.54	0.9101	0.8369	0.8066	0.7821	0.8339
		✓	2.52	0.9144	0.8382	0.8145	0.7807	0.8370
✓	✓		**1.36**	0.9196	0.8465	0.8175	0.7827	0.8416
✓		✓	2.51	0.9196	0.8458	0.8162	0.7828	0.8411
	✓	✓	2.52	0.9130	0.8387	0.8124	0.7771	0.8353
✓	✓	✓	1.46	**0.9226**	**0.8481**	**0.8227**	0.7835	**0.8442**

ITPM is evaluated both with and without the geometry supplied by PLEP. Its prototypes satisfy $p^{\pm} = \mathrm{norm}(\sum_j \pi_j^{\pm} \mathrm{norm}(f_j))$, and the per-location margin is $m_i = \cos(f_i, p^{+}) - \cos(f_i, p^{-})$.

These operations assume that the queried feature f_i and every pooled feature f_j share a spatial reference. PLEP supplies that reference by aligning coarse evidence to the unwarped shallow grid before ITPM forms p^+ and p^-. Consistent with this dependency, the standalone ITPM configuration changes the four-benchmark mean by -0.0024 relative to Base, whereas adding ITPM after PLEP changes it by $+0.0024$ in the implemented configurations. The latter comparison also includes the narrow capacity allocation and is therefore interpreted as a descriptive conditional contrast.

Along the designed Base $\rightarrow$ PLEP $\rightarrow$ PLEP+ITPM $\rightarrow$ Full path, the successive observed average-mIoU changes are $+0.0029$, $+0.0024$, and $+0.0026$. The corresponding parameter counts are 2.54M, 2.54M, 1.36M, and 1.46M. The final RPEF stage therefore adds 0.10M parameters to PLEP+ITPM for a further 0.0026 average mIoU, while the complete model remains 1.08M parameters below the reproduced baseline.

The complete 2^3 grid permits descriptive configuration contrasts. Let y_S denote the four-benchmark mean for component set S. The observed pairwise contrasts are $\Delta_{\mathrm{P}\times\mathrm{I}} = y_{\mathrm{PI}} - y_{\mathrm{P}} - y_{\mathrm{I}} + y_{\emptyset} = 0.0048$, $\Delta_{\mathrm{P}\times\mathrm{R}} = 0.0012$, and $\Delta_{\mathrm{I}\times\mathrm{R}} = 0.0007$. PLEP–ITPM has the largest descriptive contrast among the evaluated configurations. Configurations containing PLEP+ITPM also use the narrow capacity allocation, so this contrast summarizes their joint implemented configuration. The fixed-width and single-intervention controls in Table S4 separately test encoder width, prototype source, spatial offsets, residual correction, and reliability penalization.

3 Additional Fixed-Seed Results

Combining the primary seed-42 runs with seeds 2026 and 2027 gives three matched fixed-seed comparisons for DeepCrack, CamCrack789, and Crack500. Table S2 reports the two additional seeds, and Table S3 summarizes all three. FreeCrack has a positive mean paired gain on every displayed dataset. The largest gains occur on DeepCrack and CamCrack789, while Crack500 shows a smaller positive mean under the same protocol.

Table S2: Additional fixed-seed mIoU results for MixerCSeg and FreeCrack. Each row reports one recorded run for each method under the same training schedule and evaluation implementation. Gain is FreeCrack minus MixerCSeg, computed from the unrounded run records.

Seed	Dataset	MixerCSeg	FreeCrack	Gain
2026	DeepCrack	0.9116	0.9211	+0.0094
2026	CamCrack789	0.8380	0.8444	+0.0063
2026	Crack500	0.7819	0.7834	+0.0015
2027	DeepCrack	0.9117	0.9192	+0.0076
2027	CamCrack789	0.8388	0.8492	+0.0103
2027	Crack500	0.7834	0.7834	0.0000

Table S3: Three-seed mIoU summary over seeds 42, 2026, and 2027. Values are mean $\pm$ sample standard deviation across matched fixed-seed runs; paired gain is FreeCrack minus MixerCSeg for each seed before aggregation.

Dataset	MixerCSeg	FreeCrack	Paired gain
DeepCrack	0.9124 ± 0.0012	0.9210 ± 0.0017	$+0.0086 \pm 0.0010$
CamCrack789	0.8388 ± 0.0008	0.8472 ± 0.0025	$+0.0085 \pm 0.0020$
Crack500	0.7825 ± 0.0008	0.7834 ± 0.0001	$+0.0009 \pm 0.0008$

4 Targeted Controls for the Evidence Chain

Table S4 separates the complete decoder gain from encoder width and tests four mechanism-specific alternatives. With the same narrow encoder, the complete FreeCrack decoder improves all four benchmarks over the original SRF decoder and raises average mIoU by 0.0090. The complete

model also attains the highest four-benchmark mean under every targeted intervention. Relative to image-independent learned prototypes, current-image references add 0.0020 on average and 0.0073 on textured CrackMap; within RPEF, residual correction contributes 0.0034 and dispersion-based penalization supplies a further 0.0011 refinement.

Table S4: Targeted controls under matched seed-42 training. The narrow-SRF control holds the encoder width fixed and restores the original SRF decoder; each remaining control changes one mechanism of the complete FreeCrack model. Δ_{full} is the full-model average minus the corresponding control average.

Variant	Params (M)	DeepCrack	CamCrack789	CrackMap	Crack500	Avg. mIoU	Δ_{full}
Narrow encoder + original SRF	1.440	0.9110	0.8390	0.8101	0.7809	0.8353	+0.0090
Image-independent learned prototypes	1.456	0.9206	0.8485	0.8154	0.7843	0.8422	+0.0020
Zero PLEP offset	1.456	0.9168	0.8469	0.8197	0.7841	0.8419	+0.0024
RPEF residual gain fixed to zero	1.456	0.9195	0.8442	0.8146	0.7848	0.8408	+0.0034
Reliability penalty removed ($\beta = 0$)	1.456	0.9219	0.8465	0.8185	0.7853	0.8431	+0.0011
Full FreeCrack	1.456	0.9226	0.8481	0.8227	0.7835	**0.8442**	–

All control rows use the same seed, 50-epoch schedule, and evaluation implementation. The table reports the saved checkpoint associated with each completed run. The prototype-source margin is largest on textured CrackMap, consistent with the visually ambiguous regime targeted by current-image comparison. The two RPEF interventions establish an internal division of labor: the residual path supplies the main correction, while the dispersion penalty modulates the reliability of the selected evidence.

5 Cross-Domain Architectural Transfer

We evaluate the complete evidence chain as a compact decoder after adaptation to a different host architecture and image domain. Using the same DSCNet encoder and matched DRIVE training protocol, FreeCrack improves seven of nine reported segmentation metrics over the similarly compact SRF decoder. Dice increases from 0.8078 to 0.8093 and IoU from 0.6779 to 0.6799 with only 0.016M additional parameters. At 1.132M parameters, the transferred decoder retains 99.37% of native DSCNet Dice while using 45.96% fewer parameters, placing it close to the native accuracy point under a substantially smaller budget. Each decoder variant is retrained on DRIVE, so the experiment measures architectural portability across image domains.

Table S5: Cross-domain architectural transfer on the DRIVE retinal-vessel benchmark using DSCNet as the host. All metrics are produced by the same independent field-of-view evaluator at threshold 0.5. AUC_p and AUC_b denote probability and binary AUC. Bold marks the better value between the two compact replacement decoders; native DSCNet is included as the host reference.

Variant	Params (M)	Dice	IoU	Sens.	Spec.	Acc.	Prec.	AUC_p	AUC_b	clDice
Native DSCNet	2.094	0.8144	0.6871	0.8240	0.9716	0.9525	0.8094	0.9750	0.8978	0.8204
DSCNet encoder + SRF	1.116	0.8078	0.6779	**0.8061**	0.9731	0.9515	0.8141	0.9689	0.8896	**0.8105**
DSCNet encoder + FreeCrack	1.132	**0.8093**	**0.6799**	0.8057	**0.9738**	**0.9520**	**0.8179**	**0.9724**	**0.8897**	0.8102

6 Additional Qualitative Results

Figures S1–S5 and S8–S9 show matched test images, annotations, and predictions from the reproduced MixerCSeg baseline and FreeCrack. The examples from the four main benchmarks complement the distinct cases in the main paper; the TUT predictions were exported from the matched seed-42 checkpoints. The Introduction overview in the main paper uses CamCrack789 R08 for the spatial-correspondence and matched-prediction panels, CrackMap R11 for image-specific references, and DeepCrack R02 for evidence reliability; the corresponding stored inputs, labels, and

predictions are indexed below. Every prediction is displayed on the same fixed grayscale probability range, from zero (black) to one (white), without per-image contrast adjustment. These curated visual cases illustrate local behavior rather than estimate dataset-wide performance. Table S6 identifies the images and checkpoint epochs used in these panels.

Table S6: Source index for the supplementary visualizations. Checkpoints correspond to matched 50-epoch, seed-42 runs; the listed epochs identify the saved checkpoints used to generate the displayed predictions.

Dataset	Row	Test image	MixerCSeg epoch	FreeCrack epoch
DeepCrack	R02	`11296-8`	15	15
DeepCrack	R03	`IMG_6542-2`	15	15
DeepCrack	R04	`11286-2`	15	15
DeepCrack	R05	`IMG_6469-6`	15	15
CamCrack789	R06	`image-104`	32	33
CamCrack789	R07	`image-100`	32	33
CamCrack789	R08	`image-94`	32	33
CamCrack789	R10	`image-151`	32	33
CrackMap	R11	`GOPR0156 (1)`	15	22
CrackMap	R13	`GOPR0317 (1)`	15	22
CrackMap	R15	`GOPR0323 (2)`	15	22
Crack500	R16	`20160330_170848_1_721`	11	18
Crack500	R17	`20160306_104527_641_361`	11	18
Crack500	R18	`20160328_154321_1_721`	11	18
Crack500	R20	`20160307_145059_1_361`	11	18
TUT	road	`road_bPRlUrv`	22	27
TUT	road-2	`road_SIUnaBy`	22	27
TUT	pipeline	`pipeline_RPvsZAB`	22	27

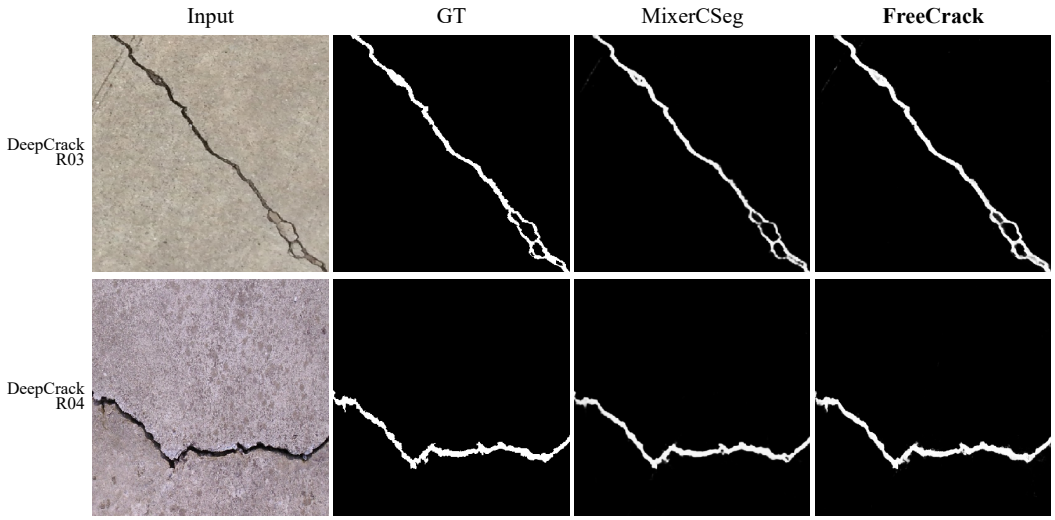

Figure S1: DeepCrack test examples R03 and R04. Each row uses the same full-frame input and annotation for both models; prediction intensities retain the fixed probability scale.

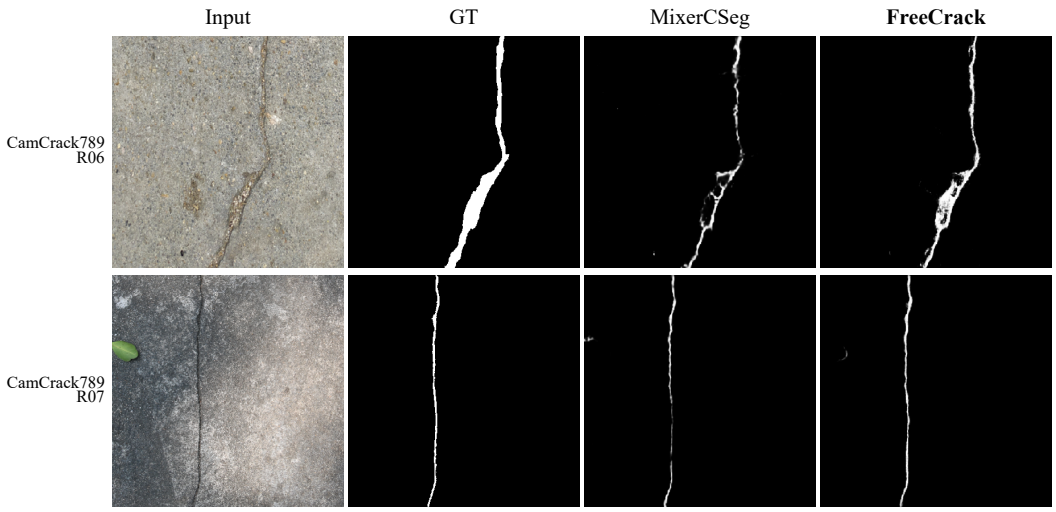

Figure S2: CamCrack789 test examples R06 and R07, including weak crack segments for which the matched predictions can be compared directly.

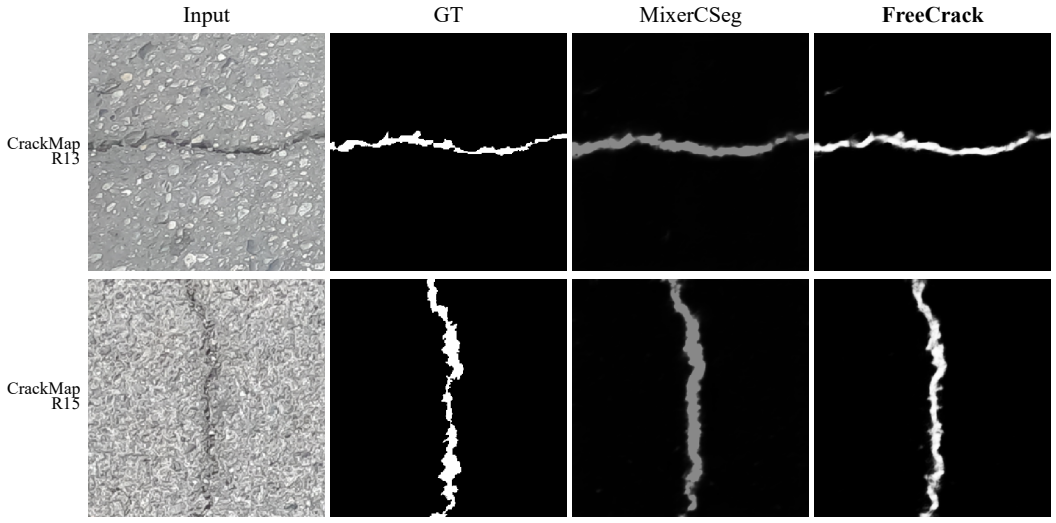

Figure S3: CrackMap test examples R13 and R15 on textured aggregate surfaces. Both crack and nearby background responses remain visible at the fixed grayscale scale.

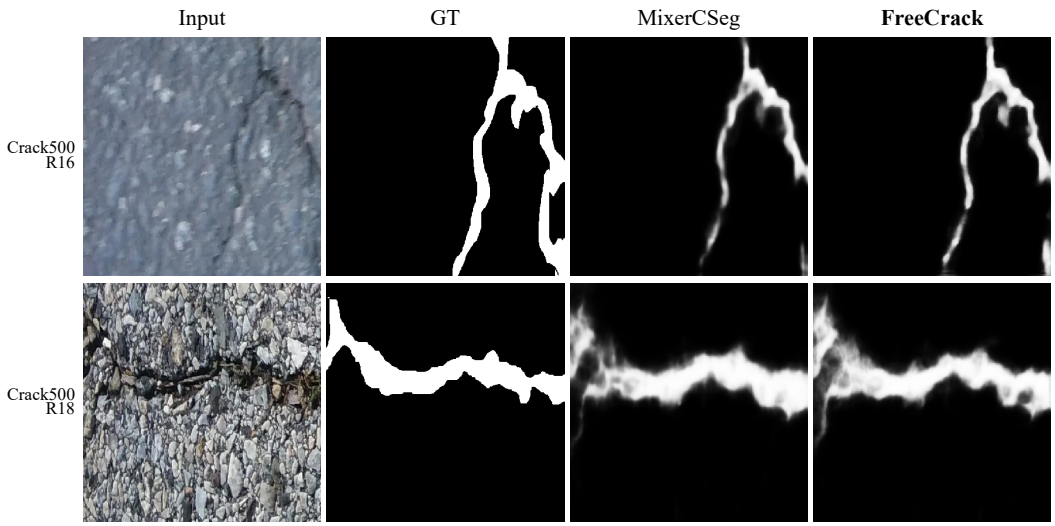

Figure S4: Crack500 test examples R16 and R18, showing curved and coarse crack structures under the same matched prediction display.

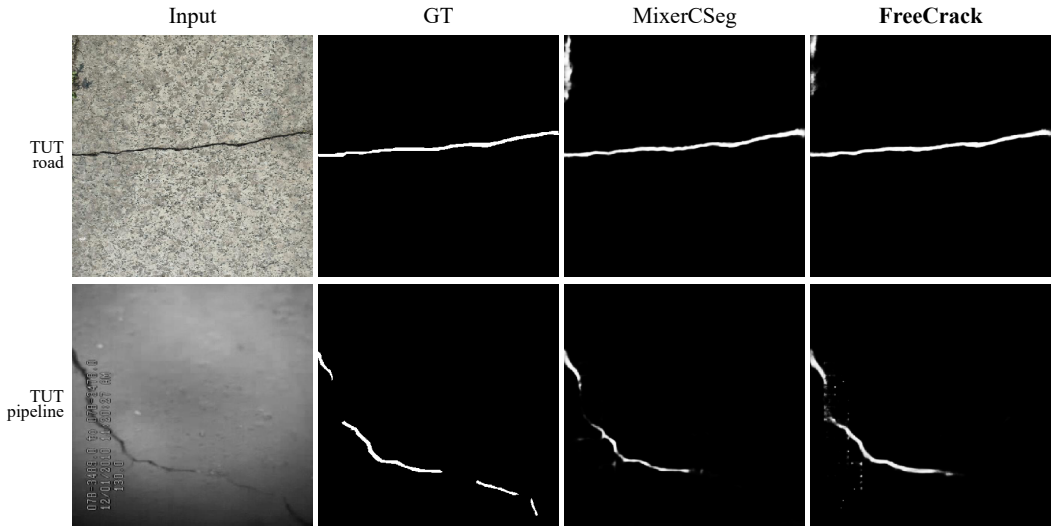

Figure S5: TUT test examples from the matched seed-42 baseline and FreeCrack checkpoints. Road and pipeline images provide two surface types under the same full-frame, fixed-probability display used above.

7 Model-Derived Evidence Maps

Figures S6 and S7 visualize tensors from the stored FreeCrack checkpoints on the same examples used in the qualitative panels. The shared-feature map is the channelwise ℓ_2 norm of PLEP feature f; the ITPM map is its foreground-minus-background cosine margin. The effective gate is the spatial gate g applied to the retained and aligned paths. The RPEF map is the query-dependent feature-dispersion penalty for the prototype-seeded token before Softmax. These are internal model tensors, not recolored segmentation predictions.

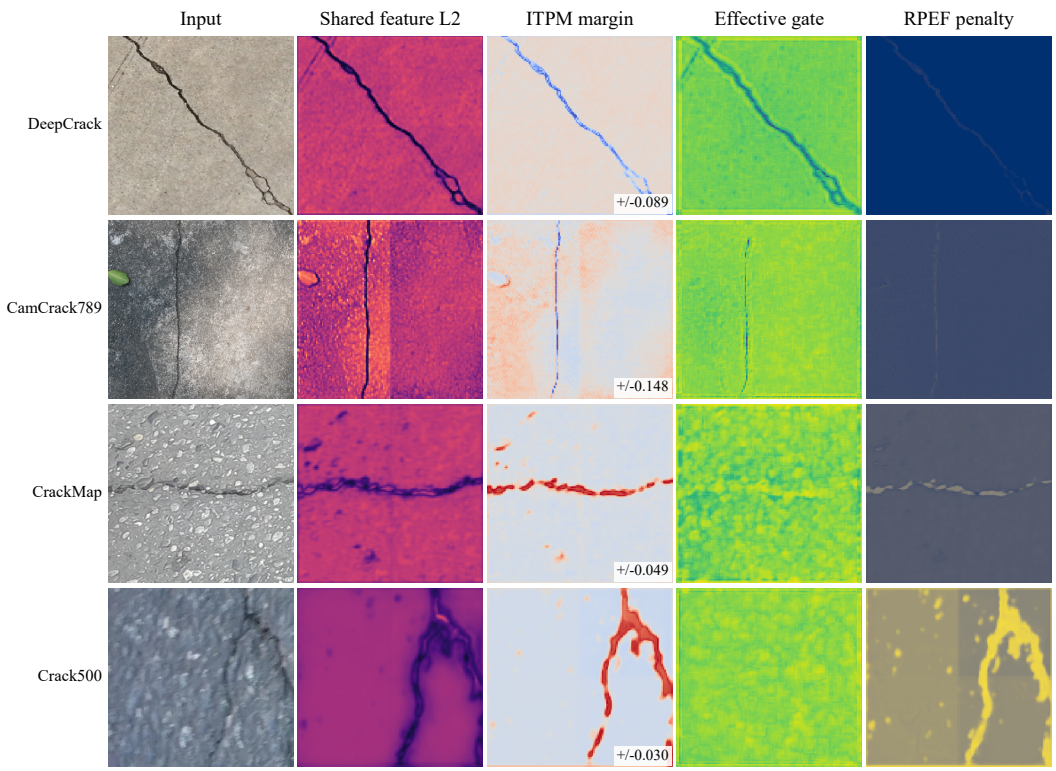

Figure S6: Internal evidence maps on DeepCrack R03, CamCrack789 R07, CrackMap R13, and Crack500 R16 (top to bottom). Feature norm, effective gate, and RPEF penalty use fixed displayed ranges $[0, 9]$, $[0, 1]$, and $[0.01, 0.065]$ across rows. Each signed ITPM margin uses the symmetric range printed in its panel. Internal maps are shown at their native 128×128 spatial grid with nearest-neighbor display.

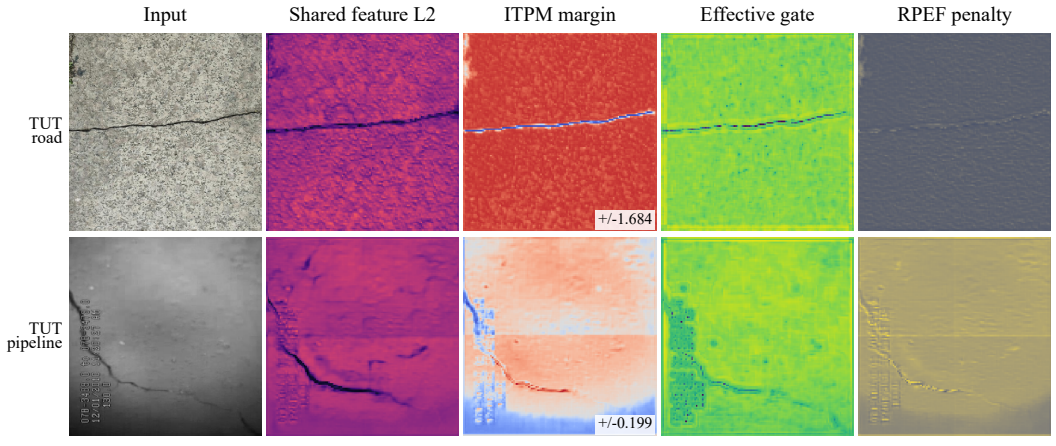

Figure S7: Internal evidence maps on the TUT road and pipeline examples (top to bottom). Tensor definitions and display ranges follow Figure S6; the per-row symmetric ITPM margin range is printed within each margin panel.

8 EXTENDED QUALITATIVE GALLERY

Figures S8 and S9 add six test images not used in the preceding supplementary panels or the main-paper qualitative figure. The matched input, annotation, and saved predictions retain the display protocol of Figures S1–S5.

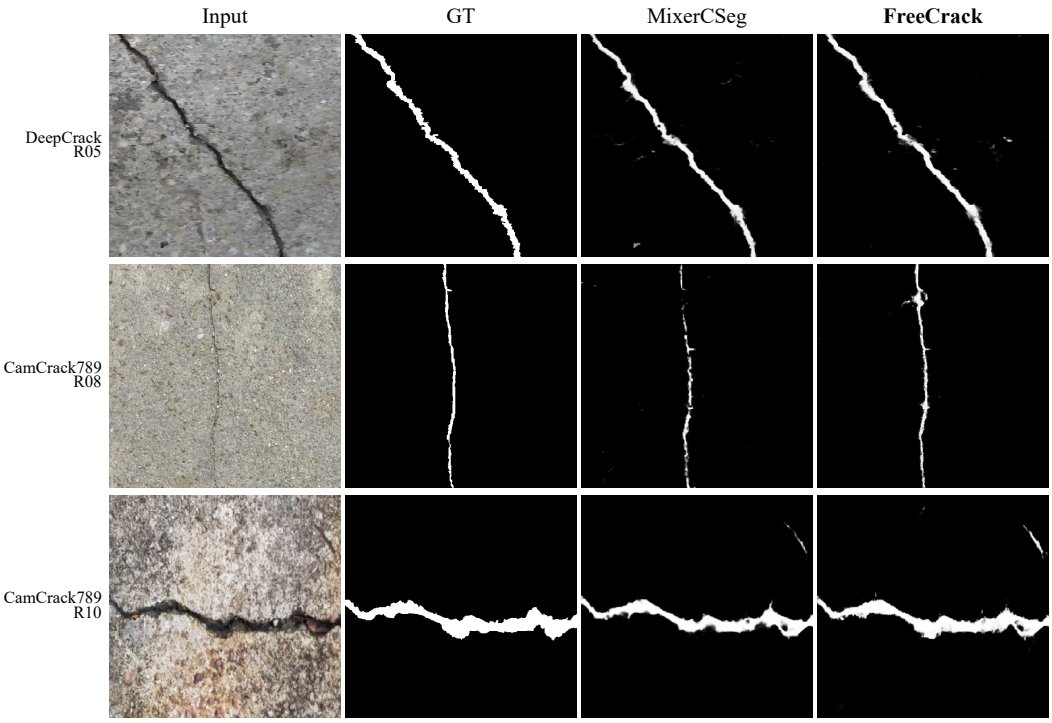

Figure S8: Additional DeepCrack and CamCrack789 cases: a narrow crack on coarse concrete (R05), a low-contrast trace (R08), and a textured region crossed by a broad crack (R10). The FreeCrack response follows the annotated structure more closely in these selected examples.

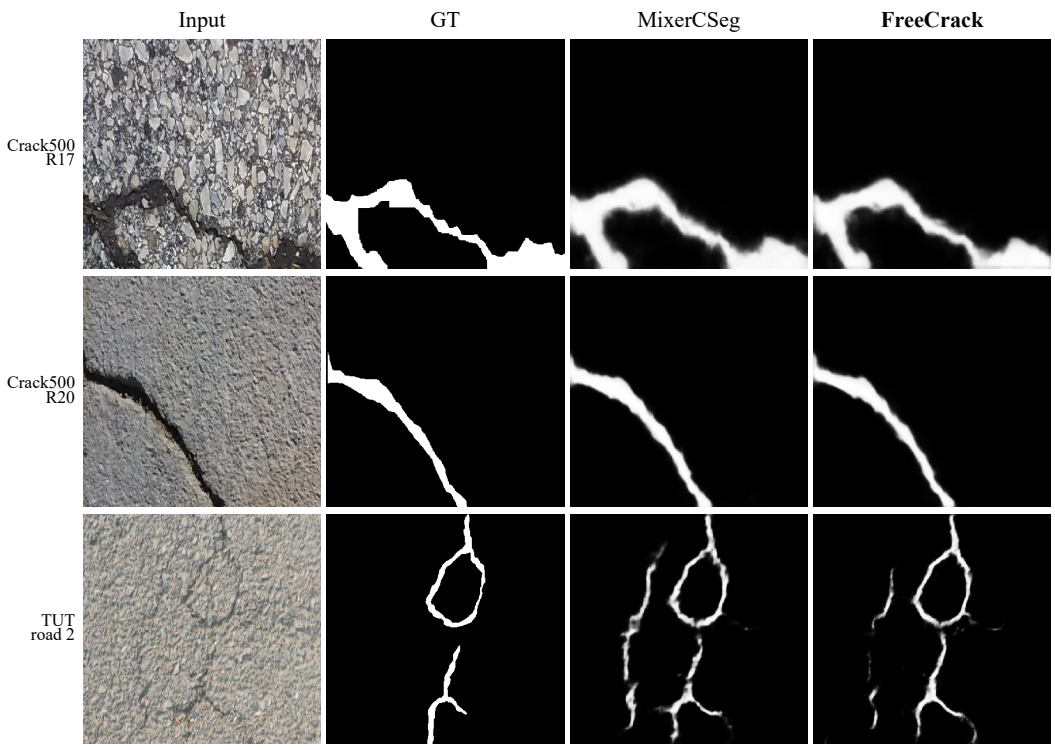

Figure S9: Additional Crack500 and TUT cases: coarse aggregate (R17), a diagonal surface crack (R20), and a branching road crack (`road_SIUnaBy`). Fixed-range grayscale predictions permit direct comparison with the matched baseline.

The saved full-frame predictions also permit an image-level check for the five main-benchmark photographs in Figures S8 and S9. Table S7 uses each photograph's complete annotation and the same 0.5 threshold; no crop is used in this calculation. The paired F1 gains span narrow, low-contrast, and aggregate-heavy crack appearances within this selected gallery.

Table S7: Full-frame F1 for the five selected main-benchmark images in Figures S8–S9. Gains use unrounded scores. These are image-level examples, not dataset averages.

Dataset	Row	MixerCSeg	FreeCrack	Gain
DeepCrack	R05	0.9003	**0.9336**	+0.0333
CamCrack789	R08	0.6158	**0.8003**	+0.1845
CamCrack789	R10	0.8584	**0.9219**	+0.0635
Crack500	R17	0.8771	**0.9130**	+0.0359
Crack500	R20	0.8828	**0.9172**	+0.0344

9 MATCHED LOCAL COMPARISONS

The following crops preserve the locations recorded in the qualitative candidate archive. Each input and annotation is placed on the same 512×512 model grid as its two saved predictions; the displayed crop is 120×120 pixels in that grid. No prediction is thresholded or contrast-normalized for display. Row labels give the FreeCrack-minus-MixerCSeg F1 difference *within the displayed crop* at a fixed threshold of 0.5. These selected-region values describe the pictured locations, not the test set.

9.1 DEEPCRACK

The five DeepCrack crops cover a narrow trunk, two junctions, and irregular crack boundaries. At the junction in R02, the additional response follows both branches. In R03, the narrow continuation remains visible through the bend. The F1 difference in each row is computed against the same annotation and crop window.

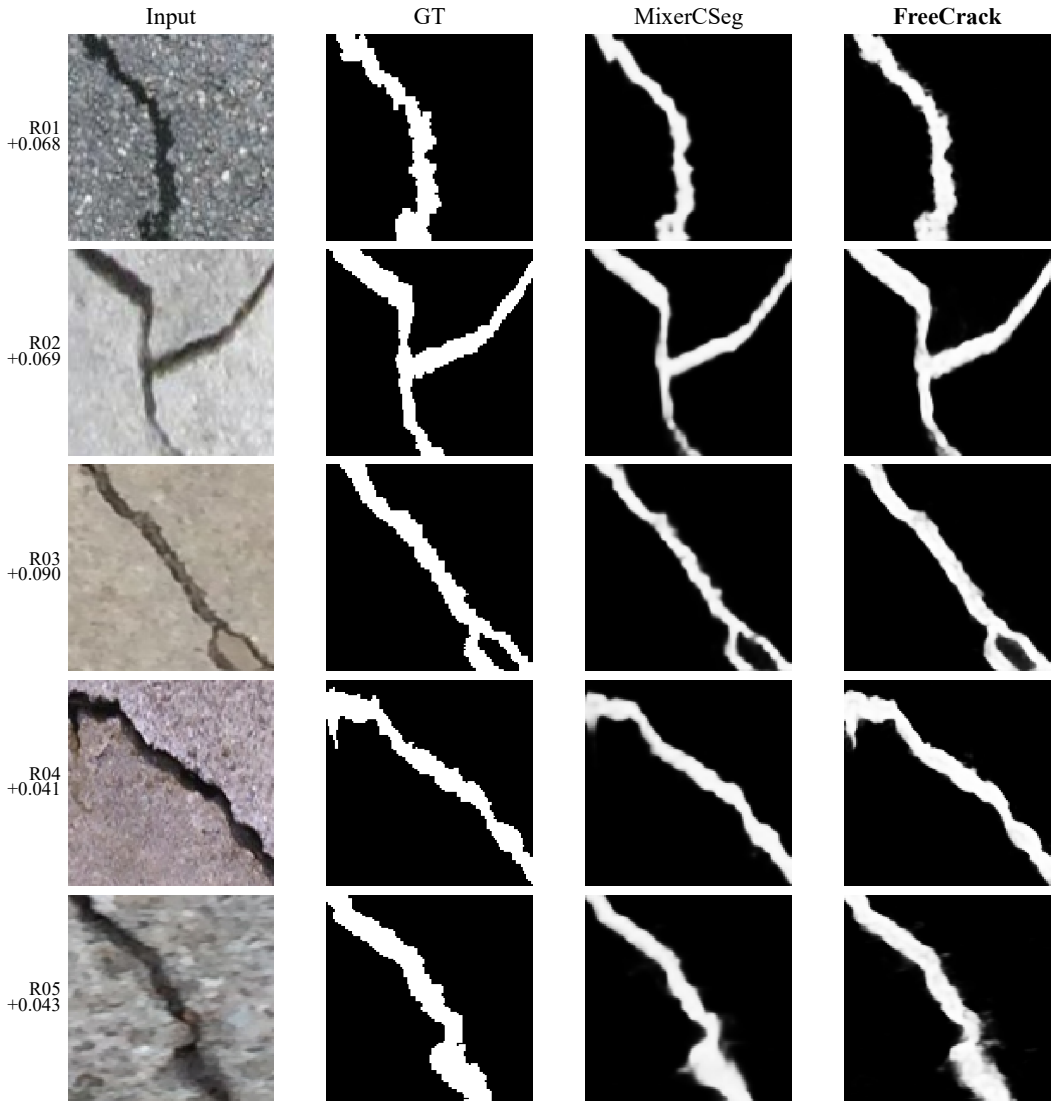

Figure S10: Matched DeepCrack crops R01–R05. The last two columns retain the same probability-to-grayscale mapping; the positive number beside each row is its selected-crop F1 difference at threshold 0.5.

9.2 CamCrack789

CamCrack789 provides two distinct visual tests. R07 and R08 contain weak, narrow traces against granular pavement; R06, R09, and R10 contain wider or curved segments. The baseline response is faint or discontinuous in the narrow cases, while the FreeCrack response extends along the annotation. The same crop coordinates are used for all four columns of a row.

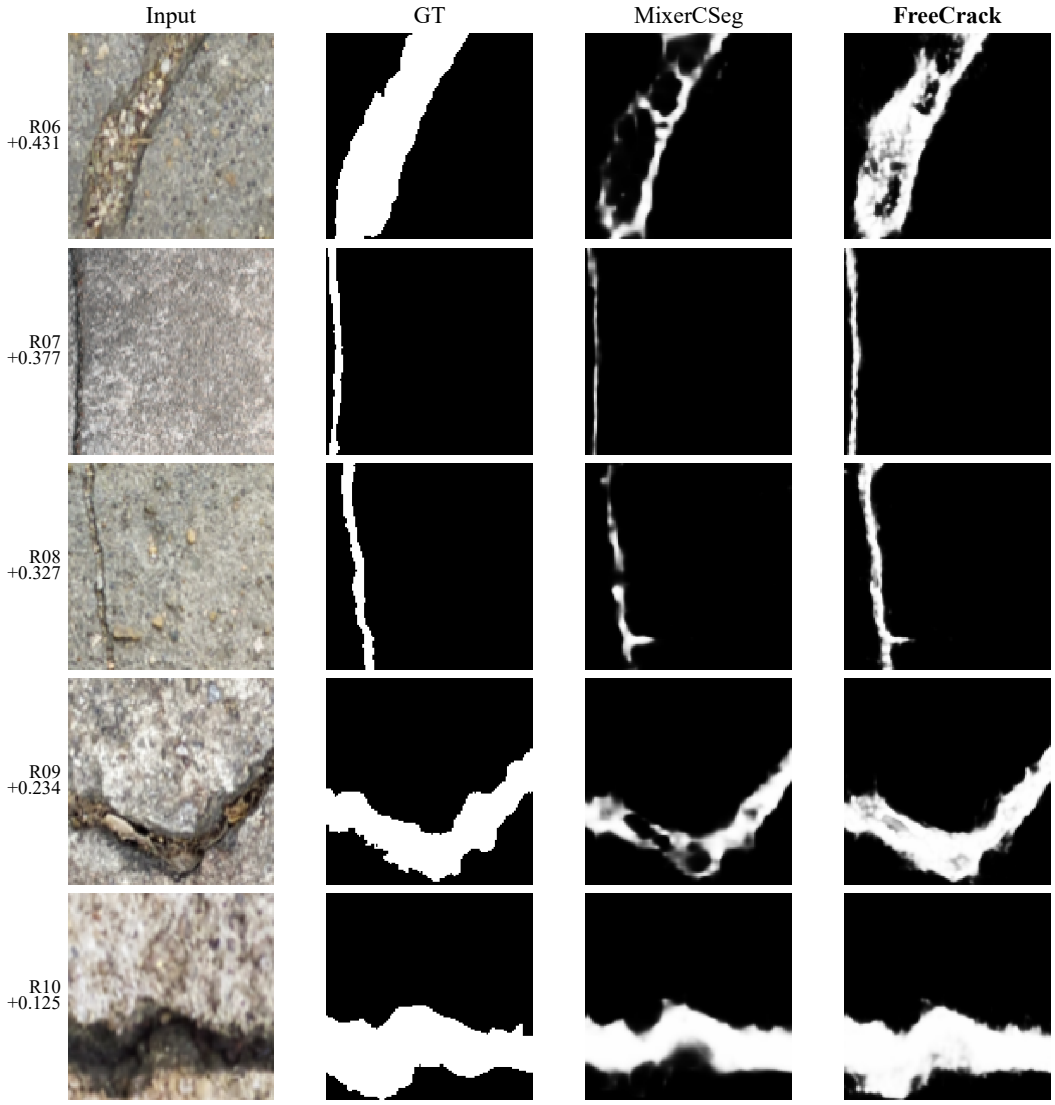

Figure S11: Matched CamCrack789 crops R06–R10. The selected windows show fine-line recovery and boundary tracking at the common, fixed display scale. Row numbers report only within-window F1 differences.

9.3 CRACKMAP

Aggregate boundaries supply many crack-like edges in CrackMap. The four pictured windows retain the surrounding texture, so the prediction can be assessed against both the annotated line and nearby non-crack structures. In R11–R14, FreeCrack concentrates a brighter response on the labeled course of the crack while reducing diffuse gray response around it.

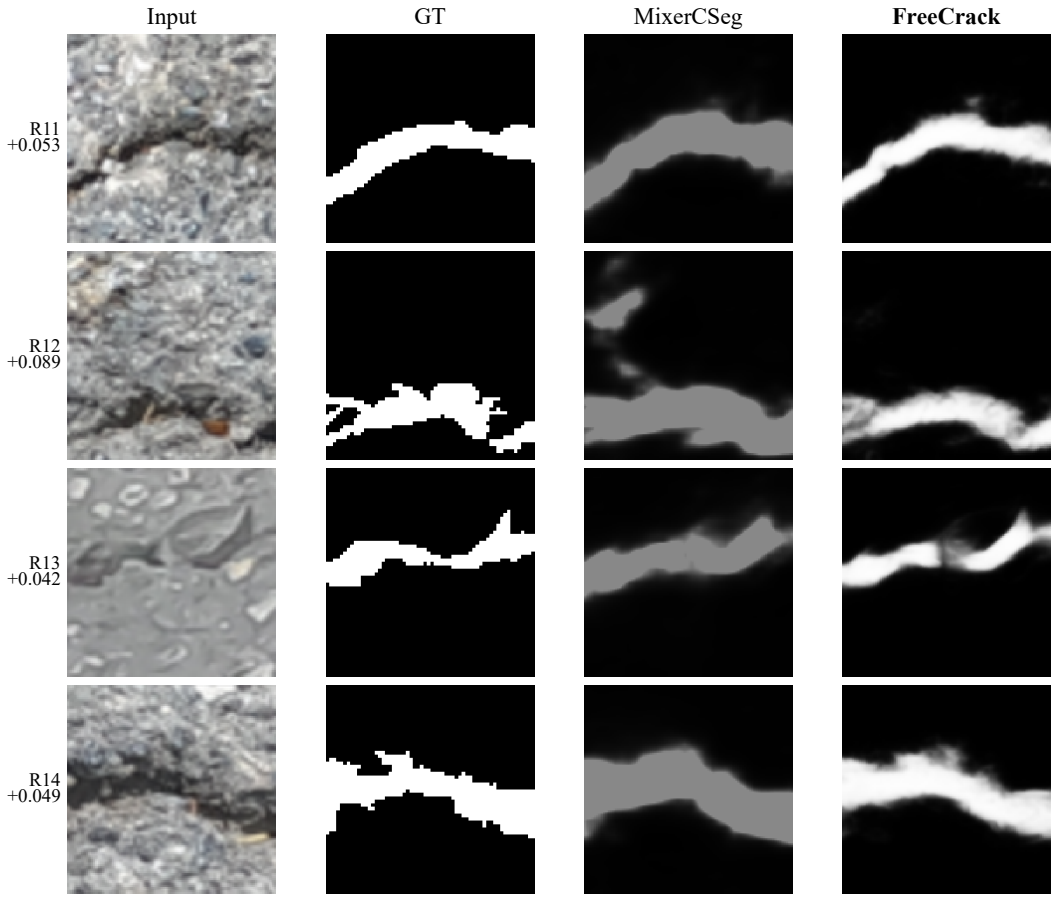

Figure S12: Matched CrackMap crops R11–R14. Input, ground truth, and both saved predictions share the normalized ROI recorded in the source archive. The grayscale prediction range is fixed across rows.

The full-frame selection scores for the same photographs also favor FreeCrack (Table S8). The local panels expose where the difference occurs; the full-frame values check that the displayed crop is not the only part of these selected images with a positive paired difference.

Table S8: Full-frame F1 at threshold 0.5 for the four pictured CrackMap test images. Gains are computed before rounding. These image-level scores describe selected examples, not the benchmark average.

Row	Test image	MixerCSeg	FreeCrack	Gain
R11	`GOPR0156 (1)`	0.7888	**0.8484**	+0.0595
R12	`GOPR0298 (1)`	0.7687	**0.8252**	+0.0564
R13	`GOPR0317 (1)`	0.7329	**0.7905**	+0.0576
R14	`GOPR0299 (3)`	0.8238	**0.8483**	+0.0246

9.4 CRACK500

The Crack500 crops include broad damaged regions, curved boundaries, and diagonal cracks embedded in coarse aggregate. The model comparison is most informative at the annotated crack edge: in R18 and R19, the FreeCrack response occupies more of the labeled interior without a corresponding expansion into adjacent texture. The remaining rows show the same positive within-window F1 relation at different crack widths.

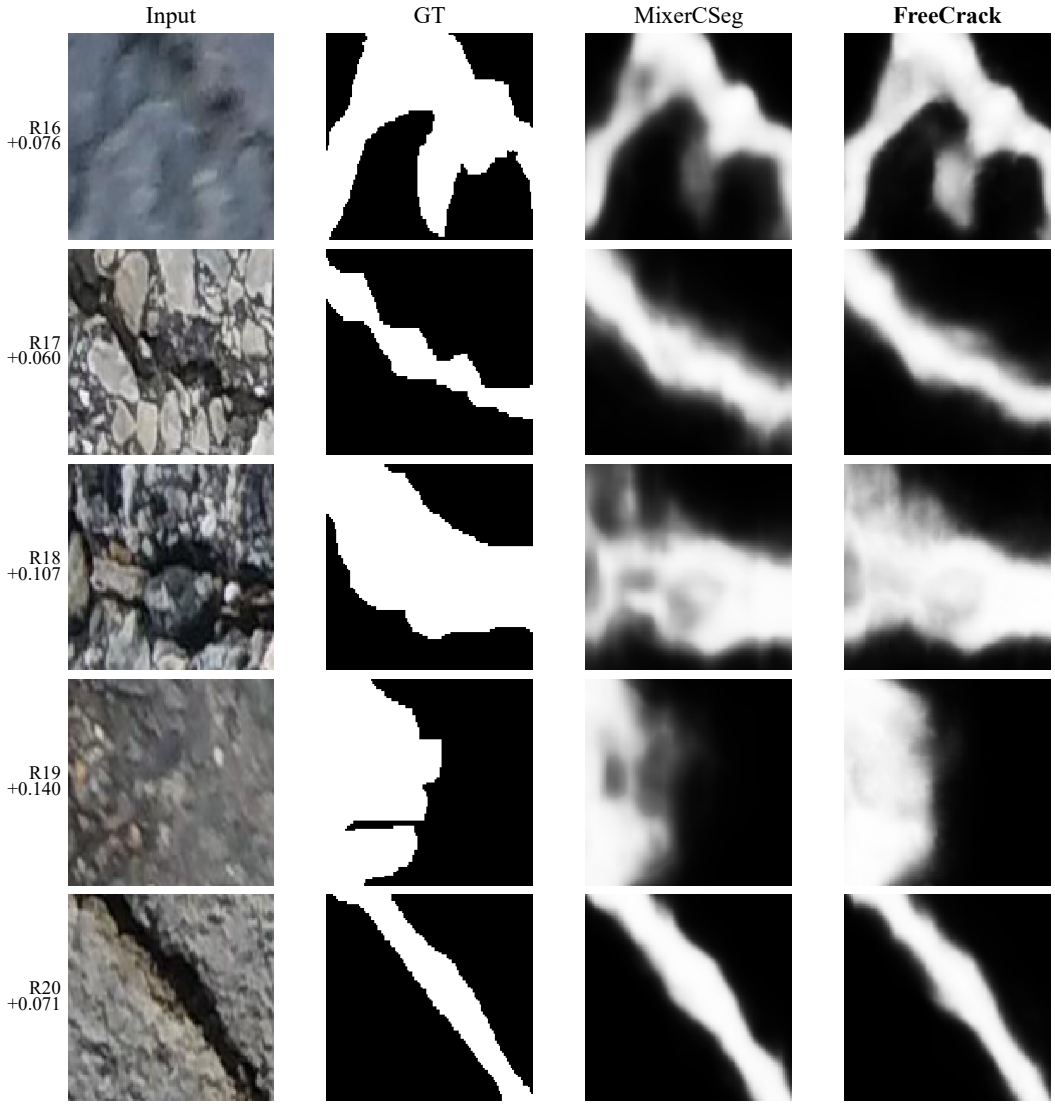

Figure S13: Matched Crack500 crops R16–R20. Each row uses a single source image, one annotation, and predictions from the matched seed-42 checkpoints. Row numbers are selected-crop F1 differences at the common 0.5 threshold.

10 Additional TUT Surface Types

The TUT gallery adds four test photographs beyond Figures S5 and S9: two road surfaces, one pipeline, and one alternator. All predictions use the matched 50-epoch, seed-42 checkpoints already indexed in Table S6. Input and annotation are displayed on the 512×512 prediction grid. Numbers beside the rows are fixed-threshold F1 differences for these selected photographs; they are not a replacement for the full TUT metrics in the main paper.

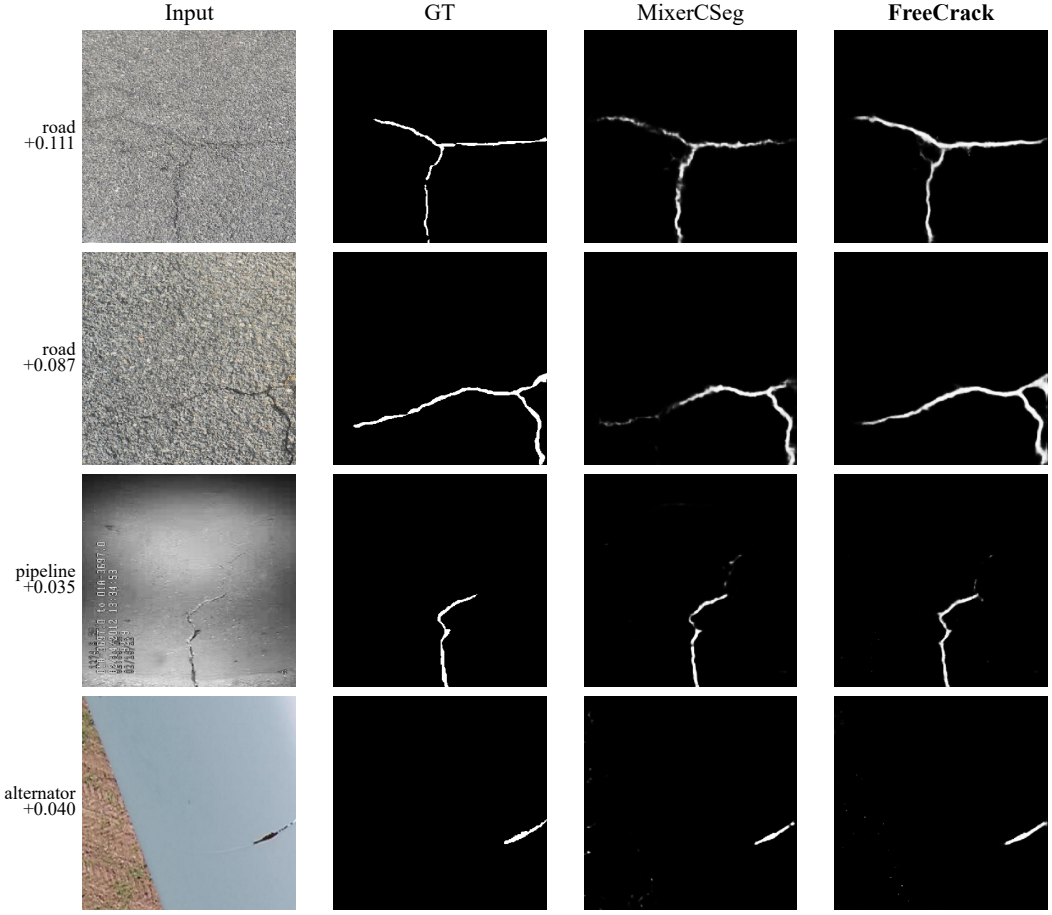

Figure S14: Four additional TUT cases, ordered as two roads, a pipeline, and an alternator. The road cases show branch continuity over dense surface texture; the other two retain the annotated line while limiting nearby response. Table S9 gives the exact test-image identifiers.

Table S9: The four test images in Figure S14, top to bottom. F1 uses the shown annotation on the 512×512 display grid and the fixed 0.5 threshold; these are selected-image values.

Test image	MixerCSeg F1	FreeCrack F1	Gain
`road_VriSljQ`	0.5527	**0.6636**	+0.1109
`road_sjjERHu`	0.6536	**0.7404**	+0.0868
`pipeline_ujADgTR`	0.8010	**0.8356**	+0.0346
`alternator_YHbepQy`	0.7987	**0.8384**	+0.0397

11 PAIRED ERROR MAPS

Figure S15 separates recovered crack pixels from background activations on three selected test photographs. Gray denotes a true positive, orange a false positive, and blue a false negative at threshold 0.5; black is true background. Both predictions in a row are compared with the same 512×512 annotation. In all three cases, both false-positive and false-negative counts move in the desired direction (Table S10). This pixel accounting makes the visual difference explicit without changing the probability displays above.

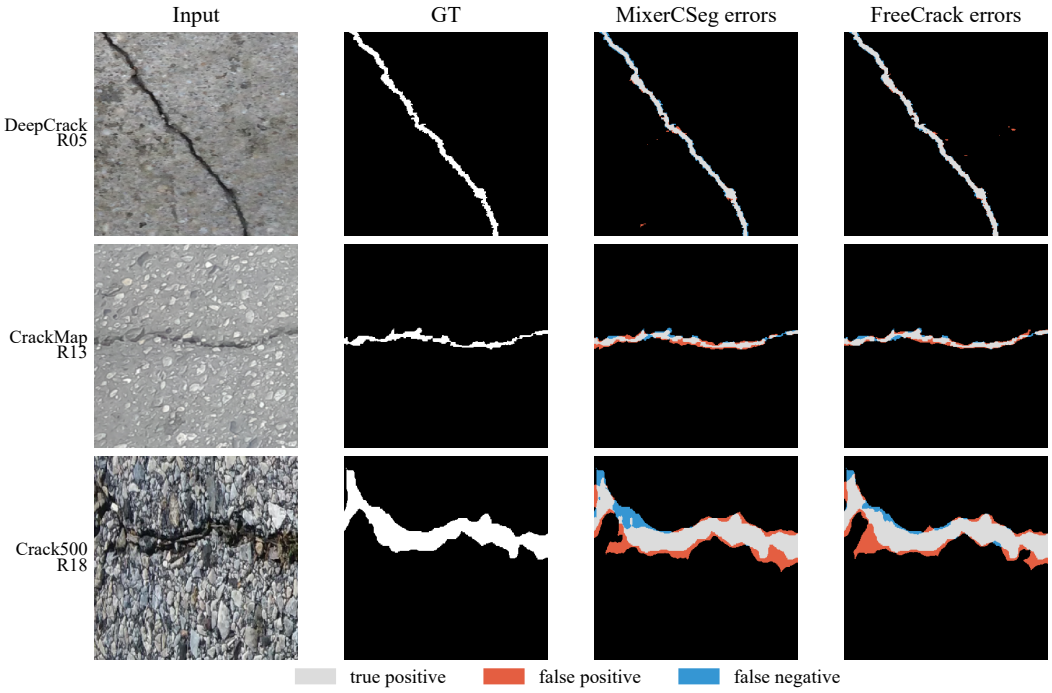

Figure S15: Matched error decomposition for DeepCrack R05, CrackMap R13, and Crack500 R18. The error colors are derived only for this diagnostic; the probability maps in Figures S8–S13 retain their original grayscale values.

Table S10: Pixel errors in the three pictured test images, using identical ground truth and a fixed 0.5 threshold. Counts refer to selected images, not dataset totals.

Dataset	Row	Baseline FP	FreeCrack FP	Baseline FN	FreeCrack FN
DeepCrack	R05	467	**450**	1,332	**745**
CrackMap	R13	2,652	**1,843**	1,037	**878**
Crack500	R18	11,067	**9,504**	4,060	**2,347**

12 Accuracy and Cost Across Component Configurations

Figure S16 joins the full four-benchmark component results in Table S1 to the measured parameter counts and DeepCrack inference latency reported for the same configurations in the main paper. Parameter count follows the implemented allocation rather than the number of enabled modules: Base, PLEP, and ITPM each use 2.54M parameters, whereas the PLEP+ITPM path uses 1.36M and the complete model uses 1.46M. The 0.10M increase from PLEP+ITPM to FreeCrack yields a further 0.0026 average mIoU. Relative to the reproduced baseline, FreeCrack adds 0.0079 mIoU while using 1.08M fewer parameters and 33.89 fewer milliseconds per image in the recorded latency setting. It is therefore the highest-accuracy point on the evaluated accuracy–size frontier. Latency is a device-specific measurement, whereas parameter count is architecture-specific.

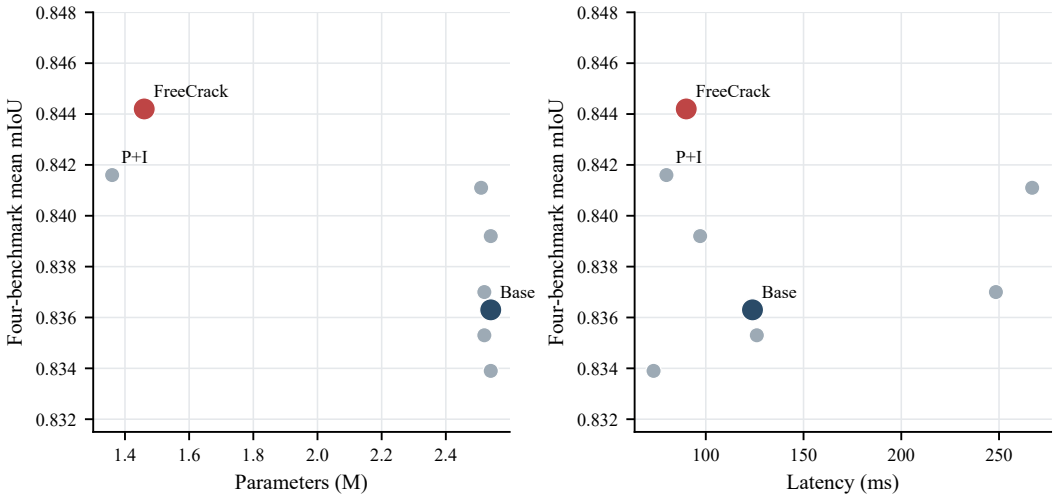

Figure S16: Mean mIoU against parameter count (left) and measured inference latency (right) for the eight evaluated component configurations. P, I, and R abbreviate PLEP, ITPM, and RPEF. Base and FreeCrack are highlighted; the remaining points are shown in gray.

The role of RPEF is clearest after the shared PLEP–ITPM representation is present. Keeping that pair fixed, adding RPEF improves mIoU on each of the four benchmarks (Table S11). The largest paired difference, 0.0052, occurs on CrackMap's textured surface. This comparison holds the alignment and image-internal reference mechanisms constant while testing the reliability-aware correction.

Table S11: Conditional contribution of RPEF to PLEP+ITPM. Values are from the matched component runs in Table S1; gains are FreeCrack minus PLEP+ITPM.

Dataset	PLEP+ITPM	FreeCrack	Gain
DeepCrack	0.9196	**0.9226**	+0.0030
CamCrack789	0.8465	**0.8481**	+0.0016
CrackMap	0.8175	**0.8227**	+0.0052
Crack500	0.7827	**0.7835**	+0.0008

13 IMAGE-SPECIFIC PROTOTYPE EVIDENCE

ITPM compares each location in the shared feature with foreground and background prototypes pooled from the same test image. Figure S17 enlarges two cases for which the saved prototype margin follows the labeled structure. The signed map is the cosine similarity to the foreground prototype minus that to the background prototype at the native 128×128 feature resolution. It is a checkpoint-derived internal tensor, not a recolored output probability. Both rows use the same displayed margin range, $[-0.05, 0.05]$, and the prediction columns use the same $[0, 1]$ grayscale range.

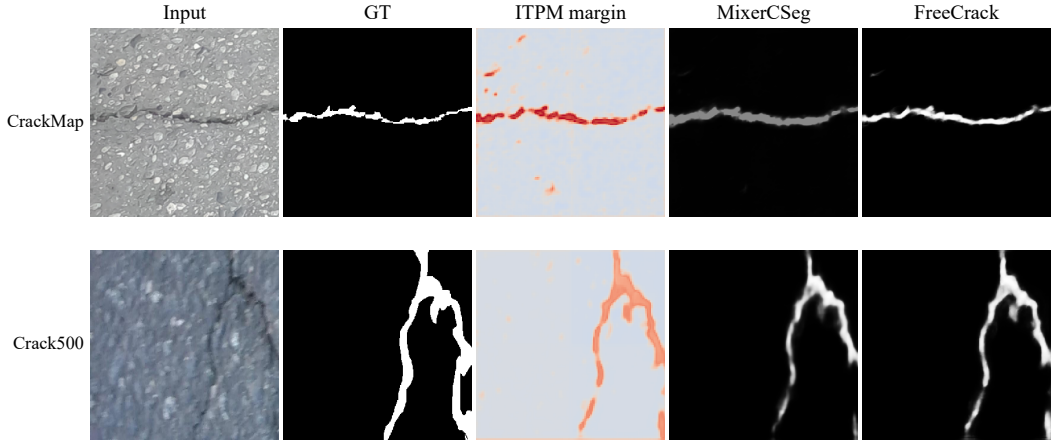

Figure S17: The stored ITPM margin on CrackMap R13 and Crack500 R16, with the matched input, annotation, and baseline/FreeCrack predictions. Red indicates a positive foreground-minus-background margin and blue a negative margin; zero is neutral.

Table S12: Mean ITPM margin within and outside the shown annotations, measured on the saved 128×128 tensor. The margin separates the two regions in these pictured cases.

Case	Annotated crack	Background	Difference
CrackMap R13	$+0.0371$	-0.0016	$+0.0387$
Crack500 R16	$+0.0162$	-0.0022	$+0.0184$

The margin enters the output as a dense logit and supplies the fourth seed for RPEF. Thus the image-internal foreground/background comparison contributes at two points in the same forward pass: directly in prediction and through the subsequent four-token evidence correction. No labeled support image or additional test-time optimization is used in either path.

The RPEF token route keeps its computation proportional to the number of image positions. At the shared feature grid, $N = 128^2$, $d = 8$, and $K = 4$ evidence tokens. Token pooling, dispersion, and query scoring scale as $O(KNd)$; the Softmax is over K tokens per location, and the pointwise residual head scales as $O(Nd^2)$. No $N \times N$ spatial attention matrix is formed. These orders describe the implemented token route, not full-network FLOPs.