Smart Structures and Systems, *Vol. 36, No. 5 (2025) 279-292*
https://doi.org/10.12989/sss.2025.36.5.279

Edge-aware transformer-based damage segmentation framework for bridge inspection maps using synthetic data

Jihun Shin [1] and Chang-Su Shim*[2]

[1] *Department of Smartcity, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul, Republic of Korea*
[2] *Department of Civil and Environmental Engineering, Chung-Ang University, 84 Heukseok-ro, Dongjak-gu, Seoul, Republic of Korea*

(*Received October 20, 2025, Revised December 9, 2025, Accepted December 17, 2025*)

Abstract. Most bridge inspection records are kept in analog formats as inspection maps with cracks annotated by hand, which limits their usefulness in modern digital asset management systems. Although deep learning has achieved strong performance on crack detection in photographic imagery, these methods depend heavily on color and texture cues that are absent in inspection maps. To address this gap, this paper presents a deep-learning framework designed to segment cracks directly from binary inspection maps. A synthetic dataset generation pipeline was developed using Auto LISP scripts to simulate cracks in CAD-based bridge elevations, reducing the need for manual annotations. Building on advances in general-purpose segmentation architectures, we design a customized model that incorporates an edge-sensitive decoding module, a structure-aware loss combining geometric and pixel-level accuracy, and a sliding-window inference strategy for processing large, high-resolution drawings. The model is trained on synthetic data and evaluated on real inspection maps to test its generalization ability. Results show that the proposed method consistently outperforms widely used segmentation baselines in both quantitative accuracy and visual clarity. Ablation studies further confirm the contribution of each architectural component. Beyond static segmentation, the framework enables time-series visualization of crack evolution, supporting condition tracking across historical records. This approach provides a scalable and practical solution for digitizing analog inspection data, making it compatible with Building Information Modeling (BIM) and digital twin systems. By transforming long-term inspection archives into actionable digital resources, the proposed method enhances efficiency, continuity, and data-driven decision-making in bridge maintenance workflows.

Keywords: crack; digitalization; inspection map; transformer

1. Introduction

The aging of bridge infrastructure has amplified the importance of adopting systematic, data-driven maintenance strategies. As of 2021, 6.8% of bridges in the United States are classified as structurally deficient while 49.1% are rated as fair, indicating that more than half of the nation's bridges require maintenance or rehabilitation (American Society of Civil Engineers 2021). Similarly, in South Korea, more than 3,500 bridges have exceeded 30 years of service life as of 2017, and over 10,000 bridges, amounting to approximately 33% of the total, are projected to be classified as aged infrastructure by 2027 (Kim *et al*. 2023). These statistics highlight the fact that aging bridge infrastructure poses a critical and increasingly urgent challenge for countries around the world.

Among these bridge deterioration issues, slabs located in the superstructure are particularly susceptible to degradation, as they are directly exposed to environmental conditions, deicing agents, and vehicular loads. When slab deterioration exceeds a certain threshold, full replacement, rather than repair or strengthening, is often required. Therefore, bridge slabs are considered as a critical component in bridge maintenance efforts (Kim *et al*. 2023).

Against this backdrop, digital-twin-based maintenance frameworks have received increasing attention as key bridge maintenance technologies (Zhou *et al*. 2022, Shim *et al*. 2019). By integrating real-time and historical data into building information models (BIM), digital twins enable predictive maintenance decision-making. However, the effective implementation of digital twin systems relies on structured time-series data that capture the long-term deterioration histories. Among the various legacy records, analog 2D inspection maps contain important information on structural deterioration but remain underutilized in practice. Therefore, digitizing analog 2D inspection maps into structured digital datasets is essential for enhancing the reliability of digital twin systems.

Traditional bridge inspection records consist of 2D inspection maps, as shown in Fig. 1. The damage types and locations have been manually annotated on these maps during close visual inspections (New York State Department of Transportation 2017). Although these analog inspection maps offer intuitive visual interpretation, they have three key limitations: (1) absence of coordinate-based quantitative data, (2) inefficient data management, and (3) incompatibility with data-driven maintenance frameworks such as digital

*Corresponding author, Ph.D., Professor,
E-mail: csshim@cau.ac.kr
[a] Graduate Student

ISSN: 1738-1584 (Print), 1738-1991 (Online)

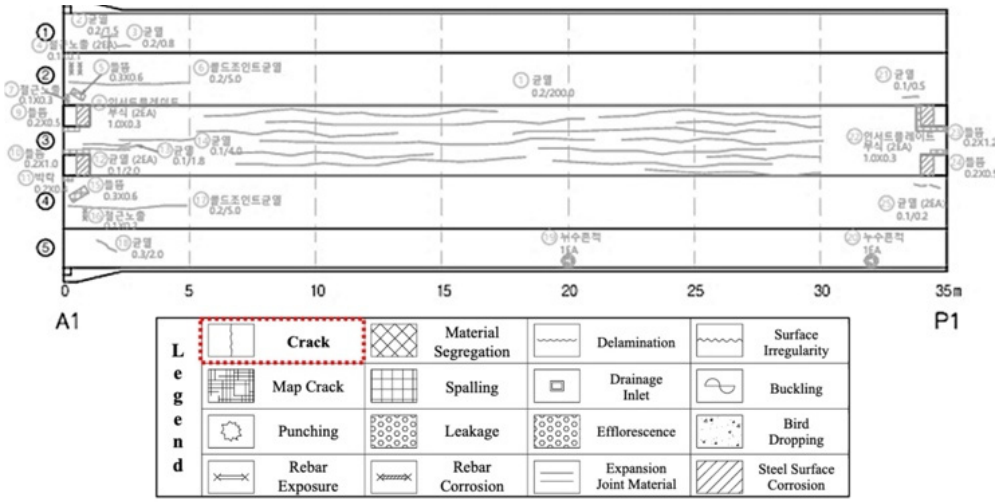

Fig. 1 Example of an analog 2D bridge inspection map with manually annotated damage information

twins. Therefore, this study proposes a data-digitization framework that automatically converts analog inspection maps into structured coordinate-based datasets to support integration with digital-twin-based maintenance systems.

A critical component of this framework is the development of a segmentation model that can identify and extract crack-like annotations from analog inspection maps. However, creating training data for a segmentation model requires the manual pixel-level segmentation of each annotation, which is extremely time-consuming and labor-intensive. Moreover, the performance of deep learning models is highly sensitive to the quantity and quality of training data (Sun *et al*. 2017, Hsu *et al*. 2022). To mitigate the lack of pixel-level labels, we build a domain-specific synthetic data pipeline that inserts crack-like line annotations into CAD-based inspection map templates and exports paired images and polygon labels in COCO (Common Objects in Context) format.

Unlike prior studies (Zou *et al*. 2019, Liu *et al*. 2021a, Wang *et al*. 2025, Bae *et al*. 2025) that detect cracks from RGB photographic images, we study the underexplored setting of binary inspection maps—CAD-like drawings with sparse, thin, and sometimes discontinuous hand-drawn line annotations amid textual and grid clutter—where photographic cues (e.g., texture, shading) are absent. Recent approaches have explored boundary refinement and multistage instance segmentation as well as super-resolution to detect ultra-thin cracks, yet these advances have been developed and validated predominantly on photographic imagery, not on binary inspection maps (Wang *et al*. 2025, Bae *et al*. 2025, Oh *et al*. 2025).

Typically, crack-like annotations on inspection maps are thin, irregular, and lack distinct visual features, making them particularly difficult to detect and segment accurately. Conventional object detection methods based on bounding boxes are useful for locating damaged regions. However, they are limited in capturing geometric attributes such as the length, area, and orientation, which are essential for tracking damage over time (He *et al*. 2017, Redmon *et al*. 2016). To overcome these limitations, this study proposes a segmentation-based approach that converts legacy inspection maps into structured coordinate-based datasets. A transformer-based segmentation model customized from the Mask2Former architecture (Cheng *et al*. 2022) was developed for this purpose. The proposed model introduces three key modifications:

(1) Edge-aware decoder module that preserves the fine geometric features of crack patterns;
(2) Structure-aware loss function combining a differentiable centerline Dice variant (soft-clDice) and Focal Loss;
(3) Sliding window inference strategy that enables the accurate segmentation of thin cracks, even in low-resolution inspection maps.

These enhancements aim to address the limitations of analog inspection maps and enable accurate and consistent digitization. By integrating the proposed components, this study developed a practical framework for automating the digitization of analog inspection maps by detecting and segmenting crack features into coordinate-based datasets. The main contributions of this study are as follows:

- A synthetic data generation algorithm tailored to inspection maps, automatically inserting crack-like line annotations into CAD-based templates to reflect binary, low-texture, line-dominated characteristics and to alleviate the lack of instance-level labels in this domain.
- A transformer-based segmentation model was designed and adapted to detect thin cracks that lack distinct visual features, by incorporating a structure-aware loss function, edge-aware decoder, and sliding window inference strategy to improve the boundary localization and segmentation quality. The digitized data from the 2D inspection map enables the quantitative assessment of damage progression between inspection years by tracking the geometric characteristics of cracks, such as their length and shape, over time. This digitized data serves as the foundational data for digital twin frameworks.
- The proposed framework generates structured,

coordinate-based crack datasets that can potentially be linked to digital twin platforms, providing the foundation for future integration with time-series monitoring or data-driven maintenance efforts.

Overall, this study contributes to the digital transformation of legacy inspection maps and offers a practical direction for the application of digital twin technologies to bridge maintenance.

2. Related work

2.1 Digital twin systems for infrastructure and the role of structured historical data

Digital twins have emerged as a promising technology for infrastructure asset management, enabling real-time monitoring, simulation, and predictive maintenance (Zhou *et al*. 2022). Building information modeling (BIM) integrates data from sensors, inspection imagery, and finite element simulations to support data-driven decision-making throughout a structure's lifecycle (Shim *et al*. 2019). Digital twin maturity is commonly classified from Level 1 to Level 5, with predictive maintenance typically requiring at least Level 3 (Jeon *et al*. 2024). In practice, however, decades of inspection records remain analog and non-standardized, impeding their integration as structured time-series for analysis or predictive workflows (Khudhail *et al*. 2021).

Recent deployments in buildings and roads show that practical decision support hinges on unified data schemas and continuously ingested, time-stamped condition histories (Chen and Brilakis 2023, Longman *et al*. 2023).

Similar principles also apply to bridge maintenance, where legacy inspection records must be converted into structured datasets. Advances in bridge information modeling (BrIM) similarly aim to structure maintenance data more systematically (Jeon *et al*. 2023), and integrating damage history, environmental conditions, and inspection data has been shown to improve maintenance decisions (Shim *et al*. 2019). Aligned with these trends, this study focuses on converting hand-annotated inspection maps into structured crack instances and coordinates for longitudinal analyses, and proposes a framework that automates the identification and segmentation of damage features in analog 2D maps for use in data-driven maintenance.

2.2 Deep learning for damage detection on binary inspection maps

Research on crack analysis has largely progressed on photographic concrete or pavement imagery. Early detection pipelines evolved into pixel-level segmentation to recover geometric details essential for maintenance decisions, while boundary-refining and multistage instance approaches improved delineation of elongated cracks (Hsieh and Tsai 2020, Wang *et al*. 2025, Bae *et al*. 2025). Super-resolution has enhanced sensitivity to ultra-thin cracks in low-resolution imagery (Oh *et al*. 2025), and transformer-based as well as lightweight architectures have achieved competitive accuracy with practical efficiency (Wang and Su 2022, Lopez Droguett *et al*. 2022). These advances characterize current practices on RGB imagery.

Inspection maps differ from photographic images, depicting damage as thin, sparse line structures with minimal texture. In such cases, models must emphasize edge geometry and linear topology rather than appearance-based contrast, making direct transfer from RGB-trained crack models suboptimal.

In this context, we review widely adopted segmentation baselines to examine their suitability for inspection maps, rather than employing crack-specific architectures. For semantic segmentation, U-Net and SegFormer represent common approaches (Ronneberger *et al*. 2015, Xie *et al*. 2021), while for instance-level segmentation, Mask R-CNN (with PointRend) and the unified Mask2Former framework are frequently used (He *et al*. 2017, Kirillov *et al*. 2020, Cheng *et al*. 2022). While crack-specific models are tailored to photographic imagery, they rely heavily on texture and shading cues absent in inspection maps. In contrast, widely used segmentation models, pre-trained on large-scale datasets, provide a more suitable baseline for evaluating segmentation strategies in this binary, line-based domain. Inspired by the unified architecture and flexibility of Mask2Former, we design a specialized model for 2D inspection maps. Our approach incorporates an edge-aware decoder to preserve fine geometric features, a structure-aware loss function that combines soft-clDice and Focal Loss, and a sliding-window inference strategy that enhances segmentation of thin cracks in low-resolution inspection maps.

3. Methodology

This paper proposes a segmentation framework designed to process inspection maps that differ from natural images because of their low visual contrast, limited color variation, and linear features. The proposed architecture is based on a state-of-the-art transformer-based model, Mask2Former (Cheng *et al*. 2022), incorporating three key innovations to enhance the segmentation performance of thin irregular crack-like objects. Fig. 2 shows an overview of the proposed segmentation pipeline. To train and evaluate this framework, a large-scale dataset with instance-level crack annotations is required. Since such data is not available for analog inspection maps, we first construct a synthetic dataset tailored to this domain.

3.1 Synthetic dataset generation

Deep learning–based object detection models are critically influenced by both the quality and quantity of the training dataset. Particularly, model accuracy tends to improve logarithmically with the dataset size (Sun *et al*. 2017, Hsu *et al*. 2022). Therefore, when training data is insufficient, synthetic data is sometimes generated to augment the data volume (Zhai *et al*. 2022). The collection of large volumes of annotated crack data from inspection maps poses considerable challenges. The majority of legacy inspection maps remain analog and unstructured, with an absence of detailed instance-level segmentation labels.

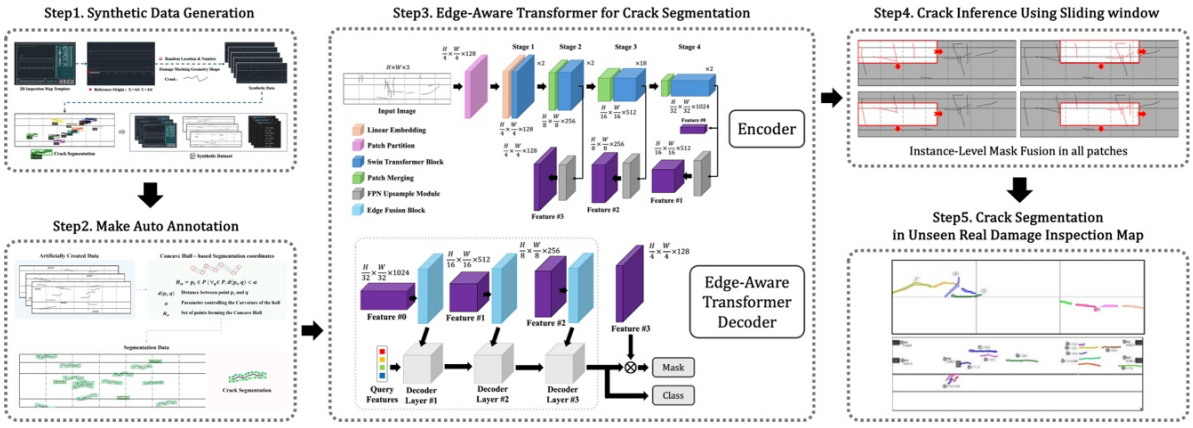

Fig. 2 Overview of the proposed framework

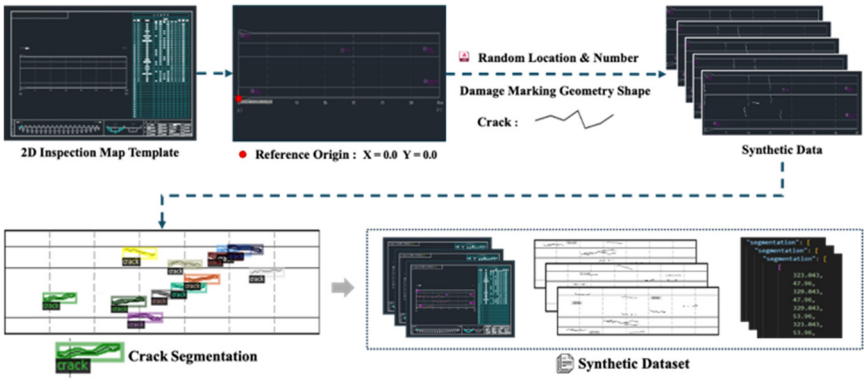

Fig. 3 Workflow of the synthetic dataset generation pipeline

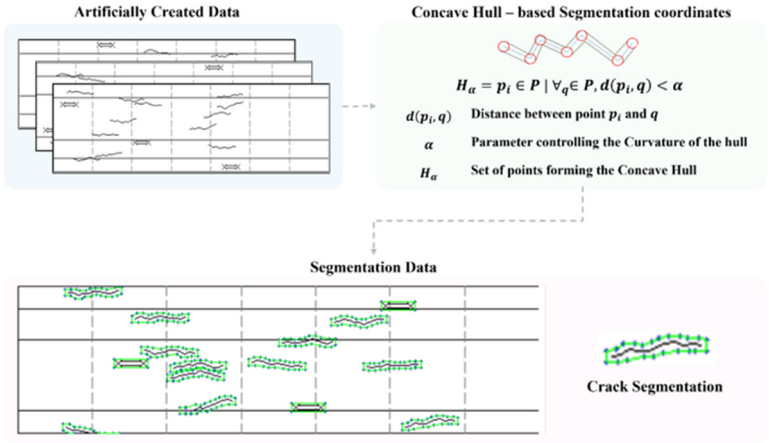

Fig. 4 Representative examples of synthetic inspection maps

Consequently, manual annotation of these datasets requires significant time and resources.

To overcome this limitation, a synthetic dataset generation framework was developed to automatically embed crack-like objects into 2D inspection map templates and generate training data in the form of paired PNG images and COCO-style JSON annotations. The entire process was scripted within the AutoCAD LISP environment. The synthesized inspection maps were first saved in the DWG format and subsequently converted into PNG images accompanied by polygonal annotations in the COCO format (Fig. 3).

3.1.1 Damage object generation

The base template for crack-like object generation is designed to emulate the layout and annotation practices of analog inspection maps used in South Korea. The insertion region is defined using the bounding coordinates that are specified relative to the reference origin and span from (X_{min}, Y_{min}) to (X_{max}, Y_{max}). Each damaged object was

generated as a polyline consisting of randomly oriented straight-line segments with properties such as the direction, length, and number of segments varying in each iteration. Crack-like-object generation involves three main steps: (1) selecting a random starting point, (2) iteratively generating line segments based on predefined angle and length ranges, and (3) extracting the polygonal boundary coordinates using a concave hull algorithm. This approach enables the generation of crack-like objects with different shapes and lengths. Each object is saved individually and reset before generating the next instance, allowing for multiple variations on a single-base template. Fig. 4 shows the representative synthetic datasets generated using this process.

3.1.2 Image generation and preprocessing

Each DWG file was exported as a PNG image at a fixed resolution of 500×150 pixels. During this process, only the portion of the map containing the damaged object was retained. This selective cropping improved training efficiency by eliminating irrelevant visual elements from the learning process. The resulting images retained the line-based structure typical of inspection maps, serving as direct inputs for the segmentation model.

3.1.3 Annotation strategy: *Bounding boxes vs. Concave-hull*

Inspection maps contain not only damaged objects but also auxiliary elements, such as grid lines and textual annotations, which may act as noise during model training. When using bounding-box-based annotations, these elements are frequently included within the labeled region, leading to reduced segmentation accuracy and model efficiency.

Table 1 Composition and usage purpose of the synthetic dataset

Set	Images	Ratio	Purpose
Train	12,311	70.0%	Used for model parameter learning with strong supervision.
Validation	3,518	20.0%	Used for hyperparameter tuning and early stopping
Test	1,759	10.0%	Used for final evaluation and ablation analysis
Total	17,588	100.0%	Full synthetic dataset for crack segmentation benchmarking

(1) Sampling pixel points uniformly along generated polyline.
(2) Generating a boundary polygon using the concave hull algorithm.
(3) Adjusting boundary granularity through the curvature parameter (α).
(4) Storing resulting polygon coordinates in COCO-compatible JSON format.

This approach improves the annotation precision by minimizing background inclusion and capturing the morphology of linear crack-like objects more accurately.

3.1.4 Dataset composition

In total, 17,588 synthetic inspection maps and annotation pairs were generated using the proposed framework. The dataset was randomly split into training, validation, and test subsets at a ratio of 70:20:10. Table 1 summarizes the dataset composition across the three subsets. algorithm

All generated data samples were verified using an automatic visualization process to confirm the label quality and detect anomalies. The dataset was organized within a structured directory hierarchy to facilitate efficient access and reuse.

3.2 Model architecture

The proposed model introduces three key components designed to improve the segmentation of thin and irregular crack-like annotations: (1) an edge-aware transformer decoder, (2) a structure-aware loss function, and (3) a sliding-window inference strategy for low-resolution inspection maps. These components address key challenges such as boundary ambiguity, foreground sparsity, and the loss of fine structural details. Table 2 summarizes the purpose of each module.

3.2.1 Edge-aware transformer decoder

Unlike conventional objects, cracks exhibit thin, irregular, and elongated patterns that often blend into the background of inspection maps. Standard transformer decoders, which rely on global attention mechanisms struggle to distinguish between subtle structures. To overcome this limitation, a modified decoder architecture, called an Edge-Aware Transformer Decoder, is proposed. This architecture explicitly incorporates edge features into the decoding process to enhance segmentation performance of crack-like objects.

Table 2 Summary of proposed modules' purpose and functionality

Component	Purpose	Description
Edge-aware decoder	Boundary localization	Injects edge intensity maps into the Transformer decoder, enhancing the model's focus on crack boundaries and elongated structures in cluttered backgrounds.
Structure-aware loss	Continuity preservation	Combines Focal Loss and soft-cl Dice to address class imbalance and preserve structural continuity of thin crack regions
Sliding inference	Fine detail recovery	Applies patch-wise prediction and IoU-based merging to reconstruct full-image results, preserving detail in low-resolution inputs.

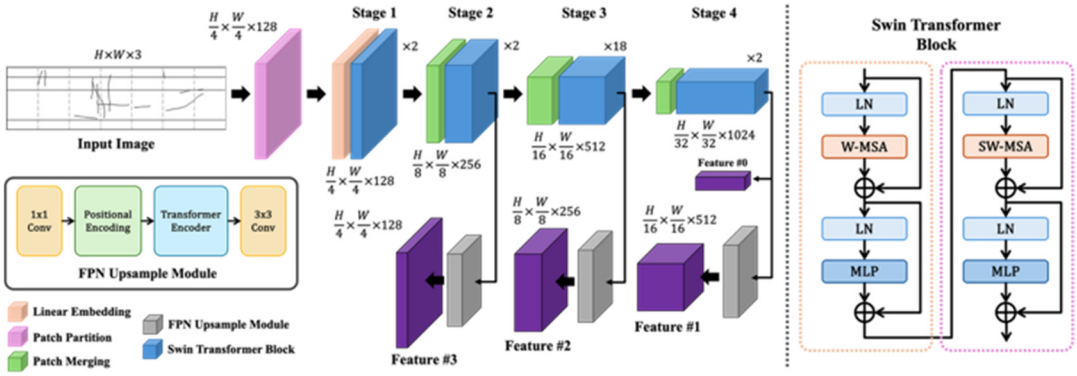

Fig. 5 Encoder architecture (modified from Cheng et al. (2022))

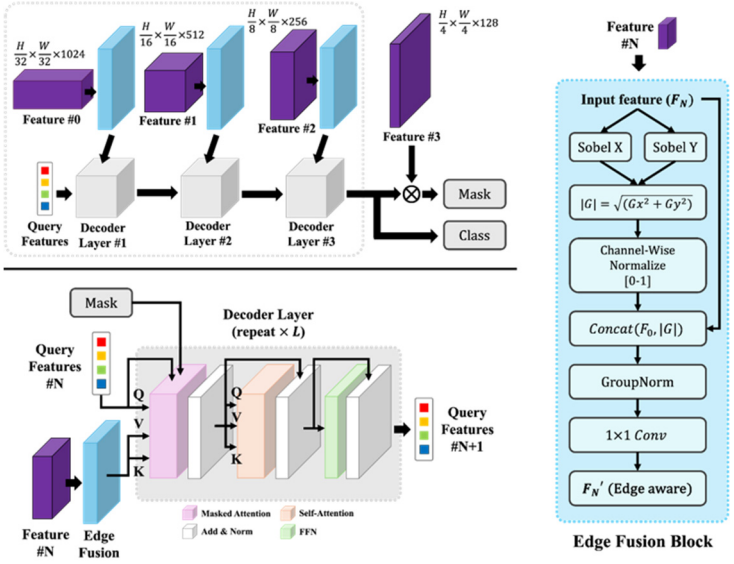

Fig. 6 Architecture of the edge-aware transformer decoder

Edge feature extraction and fusion

The multi-resolution feature map $F \in \mathbb{R}^{B\times C\times H\times W}$, is first extracted by the Swin Transformer encoder (Fig. 5), and serves as the input to the proposed edge-aware decoding process. Edge-aware features are then extracted by applying channel-wise Sobel filters (Kanopoulos *et al.* 1988) in both the horizontal and vertical directions. The Sobel kernels used for gradient computation along the x and y axes are defined as follows

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{1}$$

These kernels are convolved with each feature channel to compute the horizontal and vertical gradients. Then, the edge intensity map $E \in \mathbb{R}^{B\times C\times H\times W}$ is computed as the gradient magnitude

$$E = \sqrt{(F * K_x)^2 + (F * K_y)^2} \tag{2}$$

where * denotes the 2D convolution operator applied independently to each channel. The resulting edge map is normalized to [0, 1] range and concatenated with the original semantic feature map F along the channel dimension. This concatenated tensor is passed through a Group Normalization layer followed by a 1 × 1 convolution to produce the final edge-aware feature representation $F' \in \mathbb{R}^{B\times C\times H\times W}$

$$F' = \mathrm{Conv}_{1\times 1}(\mathrm{GN}(\mathrm{Concat}(F, E))) \tag{3}$$

This process enables the decoder to incorporate localized gradient cues while preserving semantic consistency. Edge-enhanced features improve the model's ability to capture thin and irregular crack patterns without relying on explicit boundary annotations.

Integration into transformer decoder

The overall architecture of the proposed edge-aware decoder is shown in Fig. 6. The enhanced feature F' is flattened and combined with learnable level embeddings before being processed by the decoder. Each decoder block then applies three sequential operations: multi-head cross-attention between the queries and F', multi-head self-attention among the query tokens, and a position-wise

feedforward network (FFN).

All operations are equipped with positional and query embeddings. At each decoding layer, the network outputs both class and mask embeddings. The predicted masks were obtained by computing the dot-product between the mask embeddings and the mask feature map using the Einsum operation.

Overall, the modified decoder allows the model to capture boundary-level representations more effectively, and also helps suppress irrelevant regions, thereby improving the segmentation performance of thin and irregular crack structures in high-resolution inspection maps.

3.2.2 Structure-aware loss function

Cracks on inspection maps typically exhibit thin, elongated, and irregular shapes, often with blurred boundaries that blend into the surrounding background. These characteristics pose challenges for conventional region-based loss functions, which may fail to capture structural continuity and topological fidelity.

To overcome these limitations, this study proposes a structure-aware loss function that combines Focal Loss, pixel-wise Dice Loss, and soft-clDice. Focal Loss mitigates severe foreground–background class imbalance, while Dice Loss stabilizes pixel-level overlap optimization. The soft-clDice term reinforces centerline continuity without requiring explicit skeletonization, thereby preserving differentiability during training.

All loss components were evaluated on uncertainty-guided point samples to reduce memory consumption following a point-based loss sampling framework. This integrated loss function is specifically tailored to handle the structural sparsity and boundary ambiguity commonly observed in crack segmentation tasks.

Focal loss for class imbalance

One of the primary challenges in detecting sparse objects is the extreme class imbalance between foreground and background pixels. Focal Loss (Lin *et al*. 2017) addresses this imbalance by downweighting the loss contributions of well-classified examples.

$$\mathcal{L}_{\text{focal}} = -\alpha_t (1 - p_t)^{\gamma} \log(p_t) \tag{4}$$

Where p_t is the model's estimated probability for the ground-truth class, α_t is a balancing factor (set to 1.0 in this study), and γ is a focusing parameter (set to 2.0 in our experiments). This formulation directs the attention of the model to difficult-to-classify pixels that are frequently located near thin-crack boundaries.

Soft-clDice for Structural Fidelity

To complement the region-based Dice Loss, which stabilizes pixel-wise overlap optimization, we incorporate a soft-clDice term that explicitly emphasizes structural continuity. The soft-clDice metric, inspired by clDice (Shit *et al*. 2021), quantifies the topological similarity between the predicted and ground-truth mask probability maps without explicit skeletonization. It is computed as

$$\text{soft\_clDice} = \frac{2 \mid P \odot G \mid}{\mid P \mid + \mid G \mid + \epsilon} \tag{5}$$

where P and G denote the predicted and ground-truth mask probability maps, respectively, and $\odot$ indicates element-wise multiplication.

The corresponding soft-clDice loss is defined as

$$\mathcal{L}_{\text{clDice}} = 1 - \text{soft\_clDice} \tag{6}$$

Combined structure-aware loss

The proposed structure-aware loss function is formulated as a weighted sum of Focal Loss, Dice Loss, and soft-clDice Loss, as follows

$$\mathcal{L}_{\text{total}} = \lambda_{\text{focal}} \cdot \mathcal{L}_{\text{focal}} + \lambda_{\text{dice}} \cdot \mathcal{L}_{\text{dice}} + \lambda_{\text{clDice}} \cdot \mathcal{L}_{\text{clDice}} \tag{7}$$

All three loss components contribute to preserving fine-grained structural features and alleviating foreground sparsity. Experimental results show that this combined loss function significantly improves segmentation performance for thin and irregular cracks.

3.2.3 Sliding window-based inference

Low-resolution inspection maps often suffer from visual degradation due to inconsistent scanning quality or compression artifacts, which can obscure fine crack patterns. When such images are processed using single-shot inference, early-stage feature encoding may fail to capture thin or weakly activated structures, particularly when crack segments span across larger areas or appear discontinuous within a single receptive field. To address this problem, a sliding-window inference strategy is proposed, which divides the image into overlapping localized regions to enhance crack visibility and reduce the likelihood of missed detections. This method preserves fine structural details while maintaining sufficient contextual information. The overall procedure is illustrated in Fig. 7.

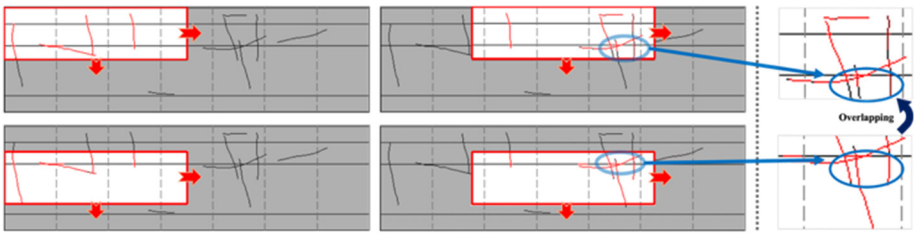

Fig. 7 Instance-level sliding window inference for low-resolution inspection maps

Patch extraction strategy

A patch size (h, w) is defined for each input image $I \in \mathbb{R}^{H\times W\times 3}$, with constraints $(h \le 0.5H, \ w \le 0.5W)$ imposed to avoid degenerate cases such as full-image inference. Overlapping patches $I_{x,y} \in \mathbb{R}^{h\times w\times 3}$ are extracted using a 50% stride along both the height and width. A 50% overlap was chosen because the cracks in inspection maps are typically thin and elongated, making them highly susceptible to fragmentation at patch boundaries. Preliminary experiments showed that smaller overlaps frequently caused discontinuities in these structures, whereas larger overlaps increased computational cost without providing additional benefits. Thus, a 50% overlap provided an effective balance for maintaining crack continuity across adjacent patches. Each predicted instance mask$m_{x,y}^{(i)} \in \{0,1\}^{h\times w}$ is reassembled into its corresponding location on the original canvas.

Instance-level mask fusion

An IoU-based merging algorithm is used to integrate overlapping predictions. Each predicted binary mask is first filtered to remove noise, such as small or nonlinear regions, and then dilated to account for slight spatial misalignments. Masks are merged if their IoU exceeds a threshold (e.g., 0.15), or if one mask substantially contains another (for example, overlaps ≥ 80% of its area). This strategy enables the effective reconstruction of fragmented cracks that span patch boundaries while preserving instance-level structure. After merging, the dilation is reversed to recover the original shape fidelity. A small constant $\varepsilon = 10^{-6}$ is added to prevent division by zero. A binary mask is obtained by applying the threshold $\tau = 0.5$ to $\hat{M}$.

These thresholds were chosen empirically based on preliminary experiments on the synthetic validation set and visual inspection of real inspection maps. IoU values lower than 0.10 often failed to reconnect partially overlapping crack fragments across patch boundaries, whereas higher thresholds above 0.20 tended to over-merge nearby but distinct cracks. Similarly, an 80% coverage criterion provided a good balance between recovering fragmented instances and avoiding spurious merging of small artifacts near patch borders, thereby maintaining continuity in the boundary regions.

Instance reconstruction

The fused masks are post-processed using geometric heuristics to eliminate false positives. Components with small area or low elongation are discarded as they represent noise or texture artifacts. Additionally, circular regions such as numbered annotation circles are detected using aspect ratio and compactness, excluded from final result. By filtering out common but non-cracked structures, the precision of instance-level predictions is improved.

The thresholds for minimum area, elongation ratio, and compactness were selected empirically based on observations from both the synthetic dataset and real inspection maps. Noise artifacts and annotation symbols typically appeared as small or compact components, whereas genuine cracks consistently exhibited larger and more elongated shapes. These geometric criteria therefore provided a reliable means of suppressing false positives. Because very small cracks can occasionally appear in real inspection maps, we also evaluated the possibility of unintentionally removing valid instances. Through qualitative inspection across multiple samples we confirmed that no meaningful crack structures were discarded. Thus, the selected thresholds offer a conservative balance between eliminating noise and preserving true crack instances. A concise flowchart summarizing the entire post-processing workflow is provided in Fig. 8.

4. Case study

4.1 Case study setup and implementation details

The proposed crack segmentation model was implemented using the PyTorch-based Detectron2 framework (Wu *et al*. 2019), with a Swin Transformer (Liu *et al*. 2021a) backbone integrated into the Mask2Former (Cheng *et al*. 2022) architecture. All experiments were conducted using a single NVIDIA RTX 3090 Ti (24GB VRAM) GPU on Ubuntu 20.04. Automatic Mixed Precision (AMP) was used to accelerate training.

The input images were synthetic inspection maps with automatically generated crack annotations, as described in Section 3. Each image was resized to a height of 145 pixels and maximum width of 500 pixels. To preserve local crack information and compensate for the low-resolution input, a sliding-window inference strategy was adopted. Each window was set to a size of 72×250 pixels with 50% overlap.

The chosen window size (72×250) corresponds to approximately half of the resized input image dimensions (145×500). This proportion was empirically found to preserve fine crack-level details while also retaining sufficient contextual information to prevent fragmented

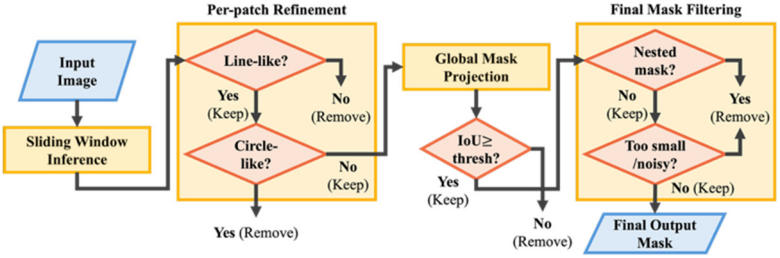

Fig. 8 Flowchart of the post-processing pipeline

Table 3 Training configuration of the proposed model

Parameter	Value
Optimizer	AdamW
Learning rate	5×10^{-5}
Warm-up	Linear (3,000 iterations)
Total iterations	30,000
Batch size	8
Weight decay	0.05
Loss weight	Focal : 2.0, Dice : 3.0, soft-clDice : 5.0, No-Object : 0.2
AMP	Enabled
Sliding window size	72×250, 50% overlap

predictions. Preliminary tests with smaller windows resulted in incomplete crack continuity, whereas larger windows reduced sensitivity to thin structures. Therefore, the selected size provided the most stable balance between resolution and context. All predictions were fused using a confidence-weighted method to produce the final binary mask. This process is illustrated in Fig. 7.

The encoder is based on a Swin Transformer with depths [2, 2, 18, 2], attention heads [4, 8, 16, 32], and a fixed window size of 12. The decoder consists of six transformer layers and 100 object queries, enabling effective learning of sparse and thin crack structures. The loss function consists of the Focal, Dice, and soft-clDice losses. The soft-clDice loss enhances topological consistency without explicit skeletonization, particularly in thin and sparsely damaged structures. To reduce the influence of background regions, the no-object class was assigned a weight of 0.2.

The training configurations are listed in Table 3. This configuration was optimized to accommodate the specific visual characteristics of inspection maps, such as low contrast, sparse structures, and fine-grained crack boundaries.

4.2 Evaluation criteria for the case study

To evaluate the performance of the proposed crack segmentation model quantitatively, this study used four standard pixel-wise evaluation metrics: precision, recall, F1-score, and IoU. Precision measures the proportion of correctly predicted crack pixels among all pixels classified as cracks, and is defined as follows.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{8}$$

where TP and FP denote the number of true-positive and false-positive pixels, respectively. Recall quantifies the proportion of actual crack pixels that are correctly identified, and is defined as

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

where FN denotes the number of false negatives. The F1-score, calculated as the harmonic mean of Precision and Recall, serves as a balanced indicator of segmentation accuracy.

$$\text{F1} - \text{score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

Finally, the IoU measures the spatial overlap between the predicted and ground-truth masks

$$\text{IoU} = \frac{TP}{TP + FP + FN} \tag{11}$$

All four metrics were computed at the pixel level and averaged over the entire test set to comprehensively assess the model's performance across cracks with varying sizes, shapes, and densities.

4.3 Datasets used in the case study

Two datasets were used under separate experimental scenarios: a synthetic dataset for supervised training and validation, and a real-world inspection map dataset used for zero-shot inference testing, where no domain-specific fine-tuning was applied. The real inspection map was obtained from the official bridge maintenance records prepared by a certified inspector. The synthetic dataset, summarized in Table 1, consisted of 17,588 paired samples generated by programmatically inserting crack-shaped polylines into structural map templates. All annotations followed the COCO polygon format, and the dataset was divided into training (70%), validation (20%), and internal testing (10%) subsets. Images were fixed at a resolution of 500×145 pixels and used without resizing, padding, or augmentation.

The real-world dataset comprised 100 diagrammatic inspection maps created by field inspectors. Manual instance-level annotations were prepared using Roboflow (Dwyer *et al*. 2025). To capture the variability of actual documents, the dataset was stratified into three difficulty levels: (i) Easy, with simple backgrounds and uniform cracks; (ii) Normal, with multiple cracks, moderate overlaps, and grid-line noise; (iii) with dense CAD layers, textual notes, and irregular crack morphologies. Representative examples are shown in Fig. 9.

This stratification provided a realistic benchmark for segmentation performance under varying levels of background clutter and crack complexity. All real-world inferences were performed using the sliding window method without resizing, serving as a zero-shot generalization test.

4.4 Baseline methods used in the case study

A comparative evaluation was conducted to assess the effectiveness of the proposed segmentation framework. The framework integrates an edge-aware decoder, structure-aware loss function, and sliding window inference strategy. Baseline models represent two categories: convolutional and transformer-based architectures. Both demonstrated strong performance in binary segmentation tasks, including crack detection and the localization of line-like damage. The selected models are the U-Net (Ronneberger *et al*.

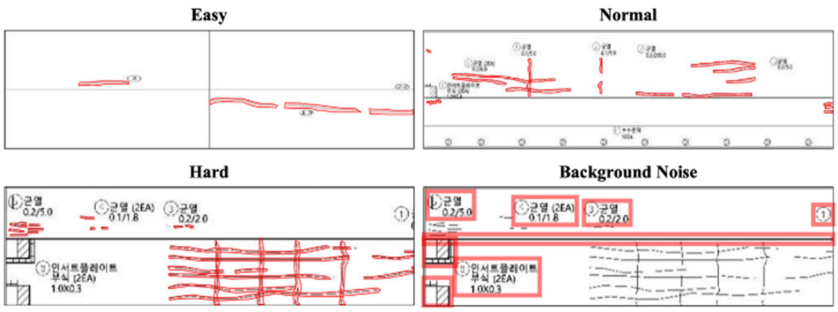

Fig. 9 Example of real inspection maps with annotated cracks at three difficulty levels

Table 4 Baseline segmentation models used for comparison

Model	Architecture type	Output type	Key characteristics
U-Net (Ronneberger *et al*. 2015)	CNN (Encoder–Decoder)	Semantic (with CCL)	Employs skip connections to preserve spatial detail; widely used in biomedical and crack segmentation tasks.
SegFormer (Xie *et al*. 2021)	Hierarchical vision transformer	Semantic (with CCL)	Utilizes multiscale attention and a lightweight decoder; robust to variations in resolution and contrast.
Mask R-CNN (He *et al*. 2017)	Two-stage instance segmentation	Instance	Predicts object masks using RoIAlign; standard baseline for instance segmentation tasks.
Mask R-CNN + PointRend (He *et al*. 2017, Kirillov *et al*. 2020)	Two-stage instance segmentation	Instance	Incorporates point-based refinement for precise boundary delineation; particularly effective for thin crack structures.
Mask2Former Cheng *et al*. 2022)	Unified transformer segmentation	Instance	Leverages mask queries and cross attention for flexible object-level segmentation; serves as the base architecture for the proposed method.

Table 5 Baseline segmentation models used for comparison

Model	Synthetic 2D Inspection Map				Real 2D Inspection Map			
	P	R	F1	IoU	P	R	F1	IoU
U-Net + CCL	**97.91**	78.37	**87.83**	**78.32**	23.79	11.85	13.36	7.8
SegFormer + CCL	91.35	75.63	82.71	70.76	14.4	7.97	8.53	4.93
Mask R-CNN	86.99	76.7	81.47	68.78	33.12	4.36	7.35	4.2
Mask R-CNN + PointRend	86.8	75.94	80.91	67.99	35.14	4.55	7.26	4.32
Mask2Former	85.74	78.49	81.88	69.36	52.51	39.82	42.68	28.32
Ours	84.32	**84.08**	84.2	72.71	**56.05**	**65.62**	**59.54**	**42.71**

2015), SegFormer (Xie *et al*. 2021), Mask R-CNN with PointRend (He *et al*. 2017, Kirillov *et al*. 2020), and Mask2Former (Cheng *et al*. 2022). The architectural characteristics are presented in Table 4.

Notably, U-Net and SegFormer were designed for semantic segmentation and inherently output a single binary mask representing the crack class. To enable a fair comparison in an instance segmentation setting, connected component labeling (Shapiro and Stockman 2001) was applied to the predicted masks as a post-processing step. While this approach allows for approximate instance separation, it fundamentally lacks the capacity to resolve overlapping or closely spaced crack structures. Consequently, these models may suffer from undersegmentation or the merging of distinct crack instances, which is a limitation considered in the evaluation and discussion of the results.

All baseline models were trained and evaluated under identical conditions, including dataset splits, input resolutions, preprocessing steps, and evaluation metrics. Optimizer configurations and training epochs were minimally adjusted to suit each model's architectural requirements. This controlled experimental setup ensured a fair and reproducible comparison in terms of boundary sensitivity, robustness to thin structural patterns, and generalization to real-world inspection maps.

5. Results and discussion

5.1 Quantitative evaluation results

Quantitative evaluations were performed on both synthetic and real inspection map datasets to validate the

performance of the proposed segmentation approach. The model's accuracy was assessed using Precision, Recall, F1-score, and IoU, as described in Section 4. As summarized in Table 5, the proposed framework exhibited competitive results on the synthetic dataset and showed clear superiority on real inspection maps.

Among the baselines, U-Net and Mask R-CNN achieved high precision on the synthetic dataset but exhibited relatively low recall, particularly for fragmented or fine cracks. However, their performance on real inspection data declined significantly, with the F1-scores falling below 15%. This highlights their limited generalization to noisy domain-shifted inputs. SegFormer effectively captured multiscale features but tended to oversegment cluttered backgrounds, reducing its overall effectiveness. Mask2Former exhibited the most balanced performance among the evaluated models, particularly for real data, but still underperformed in terms of recall and overall crack coverage compared with the proposed method.

The proposed model, which integrates an edge-aware decoder, structure-guided loss, and sliding window inference, maintained balanced performance on the synthetic dataset, while slightly trailing U-Net in precision but surpassing others in recall and IoU. On the synthetic set, it recorded an F1-score of 84.2% and IoU of 72.71%, surpassing Mask2Former by 2.32 and 3.35 percentage points, respectively.

Performance on real inspection maps—characterized by significant domain shifts and background clutter—highlighted the model's strength even more clearly, with F1 and IoU reaching 59.54% and 42.71%, improving over Mask2Former by 16.86 and 14.39 percentage points. In contrast, CNN-based baselines such as U-Net and SegFormer, even with CCL postprocessing, exhibited poor generalization, with F1-scores under 14%. These results confirm the robustness of our approach in realistic zero-shot crack segmentation scenarios.

Overall, the proposed framework demonstrated the best performance among all comparison models on the real inspection maps, proving its robustness under real inspection conditions. These results indicates that the proposed framework effectively generalizes to actual analog inspection records and provides a structured data foundation for integration with digital twin–based maintenance systems.

5.2 Quantitative comparison

In addition to the quantitative results, a qualitative comparison was conducted to evaluate the visual quality and instance-level separation of predicted crack masks. Fig. 10 shows representative segmentation results produced by the different models on real inspection maps. Notably, semantic-segmentation-based models, such as U-Net and SegFormer, output a single binary mask per class.

Despite post-processing using CCL, these methods often fail to separate overlapping or adjacent cracks, resulting in merged instances. In particular, U-Net tends to produce smooth but overly generalized contours, often distorting the actual crack geometry. Mask R-CNN-based models exhibit accurate boundary localization but often fail to detect thin or faint cracks, particularly in cluttered backgrounds.

In contrast, the proposed model generates distinct instance-level masks that retain both the sharpness and continuity of fine crack structures. By integrating a query-based decoder, edge-aware loss, and sliding window inference, the proposed method accurately delineates crack boundaries and preserves structural coherence even when

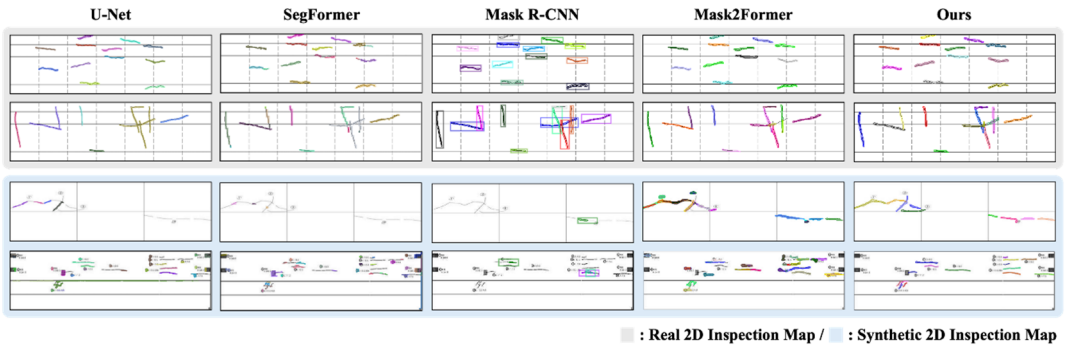

Fig. 10 Qualitative segmentation results

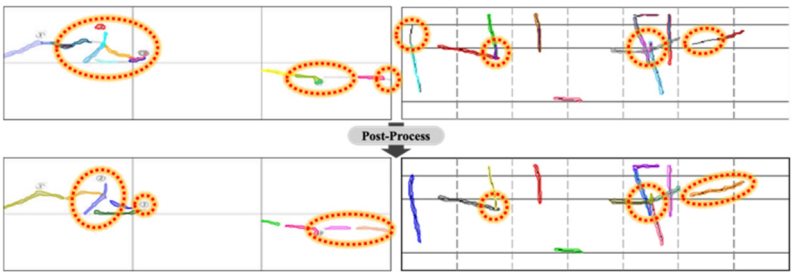

Fig. 11 Comparison of raw model predictions and post-processed results

Table 6 Effects of each component combination

Edge-aware decoder	Structure-aware loss	Sliding window inference	P	R	F1	IoU
-	-	-	52.51	39.82	42.68	28.32
O			54.3	51.07	50.44	34.82
	O		53.9	44.77	46.83	31.54
		O	47.3	53.1	48.22	32.45
O	O		58.54	58.46	56.39	39.87
O		O	49.04	68.46	55.67	39.22
	O	O	48.59	58.37	51.42	35.08
O	O	O	56.05	65.62	59.54	42.71

hand-drawn noise and CAD artifacts exist, demonstrating the model's robustness and precision in segmenting cracks under complex, domain-shifted conditions.

Through qualitative inspection of real inspection maps, we observed that each post-processing component contributes differently to the final segmentation quality. The small-area removal and circular-shape filtering were particularly effective in suppressing false positives caused by scanning noise and annotation symbols, whereas the IoU-based mask merging played a crucial role in restoring the continuity of elongated crack segments.

As shown in Fig. 11, the before/after comparison clearly illustrates these improvements: fragmented cracks become continuous, redundant predictions are consolidated, and non-crack circular artifacts are removed. These results clarify how the post-processing pipeline enhances both structural coherence and robustness of the final instance masks.

5.3 Ablation study

An ablation study was conducted to evaluate the contributions of each component within the proposed model. Starting from the Mask2Former baseline, we incrementally added three modules: (1) an edge-aware decoder that integrates Sobel-derived boundary features, (2) a structure-aware loss combining Focal Loss and soft-clDice, and (3) a sliding-window inference strategy designed for low-resolution inputs. All experiments used identical training settings and dataset.

The results in Table 6 indicate that each module contributed distinct complementary improvements. The edge-aware decoder enhanced attention to boundary-localized features, particularly under edge ambiguity conditions, leading to a 7.76% increase in F1-score. Structure-aware loss preserved the topological continuity and improved mask quality, resulting in an increase of 8.15% in the F1-score compared with the baseline. The sliding window strategy produced the highest recall gain (+13.28%), demonstrating its effectiveness in recovering spatially extended or faint cracks that are typically lost in downsampled representations.

The combination of all three modules resulted in the best overall performance, achieving an F1-score of 59.54\% and IoU of 42.71%. These results represent relative improvements of 16.86 and 14.39 percentage points in F1-score and IoU, respectively, compared with the original Mask2Former. A performance gain was achieved with only a modest increase in computational cost, demonstrating the practical feasibility of the entire architecture. The integrated approach improves structural sensitivity, boundary precision, and inference robustness, with minimal overhead.

In addition to the architectural components analyzed above, the lightweight post-processing stage also contributed to improving overall performance, particularly on real inspection maps. The small-area removal and circular-shape filtering effectively reduced false positives originating from scanning noise and annotation symbols, while IoU-based mask merging primarily enhanced the continuity of elongated crack segments across adjacent patches. Although these procedures are not part of the architectural ablation summarized in Table 6, they play a complementary role in enhancing the practical robustness of the complete pipeline.

5.4 Temporal crack progression analysis

The proposed segmentation model was applied to the 2018 and 2020 inspection maps of Bridge A to analyze temporal crack progression. The objective of this analysis was to track the growth, extension, and emergence of cracks over time and to evaluate the model's ability to support longitudinal monitoring. For each inspection map, the model produced segmentation masks of crack regions and extracted the spatial coordinates of individual instances, thereby transforming the segmentation outputs into structured, analyzable data.

As illustrated in Fig. 12, the model successfully detected the emergence of a new crack in the lower-right region that had not been present in previous inspections, and it also revealed that no significant growth was observed in the existing cracks. This temporal comparison enables alignment of crack instances across different inspection years, allowing for intuitive visualization and more precise assessment of structural deterioration trends.

In summary, the proposed method not only enables the conversion of analog inspection maps into structured digital crack data but also demonstrates practical utility in supporting time-based condition tracking and lifecycle assessments within infrastructure maintenance workflows.

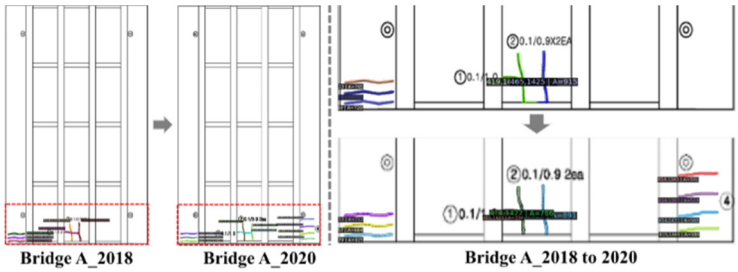

Fig. 12 Segmentation results on inspection maps from different years

Furthermore, such structured crack data hold the potential for future integration with BIM and digital twin systems, offering scalability toward long-term and systematic asset management.

6. Conclusions

This study presented a crack segmentation framework specifically tailored for inspection maps characterized by low visual contrast, sparse linear structures, and high background clutter. The proposed method is based on the Mask2Former architecture and incorporates three domain-specific enhancements: an edge-aware decoder, a structure-aware loss function, and a sliding window inference strategy.

The edge-aware decoder incorporates gradient-based boundary features to improve the delineation of thin and irregular crack contours. The structure-aware loss function combines Focal Loss and soft-clDice to address class imbalance and preserve topological continuity. The sliding window inference strategy improves the recall of faint or fragmented cracks in low-resolution images by preserving local details in overlapping regions.

Extensive experiments on both synthetic and real-world datasets confirmed the superiority of the proposed method over strong baseline models, with notable improvements in the F1-score and IoU. Ablation studies further confirmed the individual and combined contributions of each module. Preliminary experiments on time-series inspection maps confirmed the applicability of this method to longitudinal crack progression analysis. Specifically, the proposed method achieved the best performance on actual inspection datasets, demonstrating strong generalization under domain shifts and noise.

The digitized data generated in this study can serve as foundational data for integrating time-series inspection data within digital twin environments. Future research can extend the proposed framework to automate the detection of other damage types, such as rebar exposure, in inspection maps.

Data availability satement

The synthetic damage dataset-including the Auto LISP scripts used for training data generation and the custom segmentation model, which is based on a modified Mask2Former architecture, are available upon reasonable request

Acknowledgments

This research was conducted with the support of the "National R&D Project for Smart Construction Technology (RS-2020-KA156007)" funded by the Korea Agency for Infrastructure Technology Advancement under the Ministry of Land, Infrastructure and Transport, and managed by the Korea Expressway Corporation. Additionally, this research was supported by the Chung-Ang University Research Scholarship Grants in 2025.

References

American Society of Civil Engineers (ASCE) (2021), 2021 Report Card for America's Infrastructure: Bridges, American Society of Civil Engineers, Reston, VA, USA. https://infrastructurereportcard.org/cat-item/bridges-infrastructure/

Bae, S., Kim, B. and Cho, S. (2025), "Crack assessment using cascade mask R-CNN and dilation–erosion processing technique", *J. Comput. Civil Eng.*, **39**(5), p. 04025054. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001101

Chen, W. and Brilakis, I. (2023), "Developing digital twin data structure and integrated cloud digital twin architecture for roads", *Comput. Civil Eng.*, pp. 424-432. https://doi.org/10.1061/9780784484470.053

Cheng, B., Misra, I., Schwing, A.G., Kirillov, A. and Girdhar, R. (2022), "Masked-attention mask transformer for universal image segmentation", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1290-1299. https://doi.org/10.1109/CVPR52688.2022.00136

Dwyer, B., Nelson, J., Hansen, T. and Roboflow Team (2025), "Roboflow (version 1.0) [software]", Computer Vision Platform, Roboflow, USA. https://roboflow.com

He, K., Gkioxari, G., Dollár, P. and Girshick, R. (2017), "Mask R-CNN", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961-2969. https://doi.org/10.1109/ICCV.2017.322

Hsieh, Y.A. and Tsai, Y.J. (2020), "Machine learning for crack detection: Review and model performance comparison", *J. Comput. Civil Eng.*, **34**(5), p. 04020038. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000910

Hsu, S.-H., Chang, T.-W. and Chang, C.-M. (2022), "Impacts of label quality on performance of steel fatigue crack recognition using deep-learning image segmentation", *Smart Struct. Syst., Int. J.*, **29**(1), 207-220. https://doi.org/10.12989/sss.2022.29.1.207

Jeon, C.H., Nguyen, D.C., Roh, G. and Shim, C.S. (2023), "Development of BRIM-based bridge maintenance system for

existing bridges", *Buildings*, **13**(9), p. 2332. https://doi.org/10.3390/buildings13092332

Jeon, C.H., Seo, D.W., Park, S. and Park, K.T. (2024), "Review of digital twin research trends for bridge maintenance", *J. Korean Soc. Disaster Secur.*, **17**(4), 51-62. https://doi.org/10.54558/kosds.2024.17.4.51

Kanopoulos, N., Vasanthavada, N. and Baker, R.L. (1988), "Design of an image edge detection filter using the Sobel operator", *IEEE J. Solid-State Circuits*, **23**(2), 358-367. https://doi.org/10.1109/4.996

Khudhail, A., Li, H., Ren, G. and Liu, S. (2021), "Towards future BIM technology innovations: A bibliometric analysis of the literature", *Appl. Sci.*, **11**(3), p. 1232. https://doi.org/10.3390/app11031232

Kim, K.H., Nam, M.S., Hwang, H.H. and Ann, K.Y. (2020), "Prediction of remaining life for bridge decks considering deterioration factors and propose of prioritization process for bridge deck maintenance", *Sustainability*, **12**(24), p. 10625. https://doi.org/10.3390/su122410625

Kim, H.J., Seong, Y.H., Han, J.W., Kwon, S.H. and Kim, C.Y. (2023), "Demonstrating the test procedure for preventive maintenance of aging concrete bridges", *Infrastructures*, **8**(3), p. 54. https://doi.org/10.3390/infrastructures8030054

Kirillov, A., Wu, Y., He, K. and Girshick, R. (2020), "PointRend: Image segmentation as rendering", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9799-9808. https://doi.org/10.1109/CVPR42600.2020.00981

Lin, T.Y., Goyal, P., Girshick, R., He, K. and Dollár, P. (2017), "Focal loss for dense object detection", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980-2988. https://doi.org/10.1109/ICCV.2017.324

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S. and Guo, B. (2021a), "Swin transformer: Hierarchical vision transformer using shifted windows", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012-10022. https://doi.org/10.1109/ICCV48922.2021.00981

Liu, H., Miao, X., Mertz, C., Xu, C. and Kong, H. (2021b), "Crackformer: Transformer network for fine-grained crack detection", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3783-3792. https://doi.org/10.1109/ICCV48922.2021.00379

Longman, R.P., Xu, Y., Sun, Q., Turkan, Y. and Riggio, M. (2023), "Digital twin for monitoring in-service performance of post-tensioned self-centering cross-laminated timber shear walls", *J. Comput. Civil Eng.*, **37**(2), p. 04022055. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001031

Lopez Droguett, E., Tapia, J., Yanez, C. and Boroschek, R. (2022), "Semantic segmentation model for crack images from concrete bridges for mobile devices", *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, **236**(4), 570-583. https://doi.org/10.1177/1748006X221082255

New York State Department of Transportation (2017), *Bridge Inspection Manual*, New York State Department of Transportation, Albany, NY, USA.

Oh, D., Jeong, S., Bae, S., Kim, B. and Cho, S. (2025), "Training deep learning segmentation models using super-resolution crack images for detection of thin concrete cracks", *J. Comput. Civil Eng.*, **39**(4), p. 04025035. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001095

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016), "You only look once: Unified, real-time object detection", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788. https://doi.org/10.1109/CVPR.2016.91

Ronneberger, O., Fischer, P. and Brox, T. (2015), "U-Net: Convolutional networks for biomedical image segmentation", In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, Munich, Germany, pp. 234-241. https://doi.org/10.1007/978-3-319-24574-4_28

Shapiro, L.G. and Stockman, G.C. (2001), *Connected Component Labeling and Applications*, Prentice Hall, Upper Saddle River, NJ, USA.

Shim, C.S., Dang, N.S., Lon, S. and Jeon, C.H. (2019), "Development of a bridge maintenance system for prestressed concrete bridges using 3D digital twin model", *Struct. Infrastruct. Eng.*, **15**(10), 1319-1332. https://doi.org/10.1080/15732479.2019.1604772

Shit, S., Paetzold, J.C., Sekuboyina, A., Mateus, D., Eisenmann, M., Brugnara, G., Hering, L., Isensee, F., Maier-Hein, K.H., Menze, B. and Tetteh, G. (2021), "clDice: A differentiable metric and loss function for tubular structure segmentation", *Proceedings of IEEE/CVF Conference on Computer Vision Pattern Recognittion*, pp. 1652-1661. https://doi.org/10.1109/CVPR46437.2021.00170

Sun, C., Shrivastava, A., Singh, S. and Gupta, A. (2017), "Revisiting unreasonable effectiveness of data in deep learning era", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 843-852. https://doi.org/10.1109/ICCV.2017.97

Wang, W. and Su, C.E. (2022), "Automatic concrete crack segmentation model based on transformer", *Autom. Constr.*, **139**, p. 104275. https://doi.org/10.1016/j.autcon.2022.104275

Wang, C., Liu, H., An, X., Gong, Z. and Deng, F. (2025), "DCNCrack: Pavement crack segmentation based on large-scaled deformable convolutional network", *J. Comput. Civil Eng.*, **39**(2), p. 04025009. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001071

Wu, Y., Kirillov, A., Massa, F., Lo, W.Y. and Girshick, R. (2019), "Detectron2: A modular object detection library", Facebook AI Research Report. https://github.com/facebookresearch/detectron2

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M. and Luo, P. (2021), "SegFormer: Simple and efficient design for semantic segmentation with transformers", *Adv. Neural Inf. Process. Syst.*, **34**, 12077-12090. https://doi.org/10.48550/arXiv.2105.15203

Zhai, G., Narazaki, Y., Wang, S., Shajihan, S.A.V. and Spencer, B.F. (2022), "Synthetic data augmentation for pixel-wise steel fatigue crack identification using fully convolutional networks", *Smart Struct. Syst., Int. J.*, **29**(1), 237-250. https://doi.org/10.12989/sss.2022.29.1.237

Zhou, C., Xiao, D., Hu, J., Yang, Y., Li, B., Hu, S., Demartino, C. and Butala, M. (2022), "An example of digital twins for bridge monitoring and maintenance: Preliminary results", *International Conference of the European Association on Quality Control of Bridges and Structures* (*EUROSTRUCT 2021*), Springer, Munich, Germany, pp. 1134-1143. https://doi.org/10.1007/978-3-030-91877-4_123

Zou, Q., Zhang, Z., Li, Q., Qi, X., Wang, Q. and Wang, S. (2019), "DeepCrack: Learning hierarchical convolutional features for crack detection", *IEEE Trans. Image Process.*, **28**(3), 1498-1512. https://doi.org/10.1109/TIP.2018.2878963