This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

CrackSSM: Reviving SSMs for Crack Segmentation via Dynamic Scanning

Yubin Gu[1,†], Boyang Hou[1,†], Yuan Meng[1], Wenting Luo[1], Jiayi Ji[1,2], Xiaoshuai Sun[1,*]
[1]Key Laboratory of Multimedia Trusted Perception and Efficient Computing,
Ministry of Education of China, Xiamen University, 361005, P.R. China
[2]National University of Singapore, Singapore
[†]These authors contributed equally to this work. [*]Corresponding author.
{guyubin,37220222203614}@stu.xmu.edu.cn, xssun@xmu.edu.cn

Abstract

Crack segmentation (CS) is crucial for structural inspection and maintenance in production scenarios. To achieve both high accuracy and efficiency, recent methods have adopted Mamba-based architectures built upon state space models (SSMs), which enable linear-complexity modeling of long-range dependencies. However, existing approaches typically rely on static multi-directional scanning to flatten visual features into sequences. This fixed flattening order disrupts spatial continuity and weakens the SSM's ability to model irregular crack patterns effectively. To address this limitation, we propose ***CrackSSM****, a novel crack-aware segmentation framework featuring a dynamic scanning strategy that adapts the token sequence to the underlying structure of each image. Specifically, we compute directional response strength along four orientations from high-level semantic features, and use these values to reorder tokens so that crack-relevant regions remain adjacent in sequence. This alignment improves the causal modeling ability of SSMs while preserving their efficiency and better suits the irregular, fine-grained nature of cracks. Additionally, we design a wavelet-guided decoding mechanism to recover detailed features. It incorporates high-frequency components extracted from the input image and applies them to guide feature refinement and edge-aware fusion, further enhancing segmentation precision. Experiments on three benchmark datasets demonstrate that our method achieves superior segmentation accuracy with fewer parameters and faster inference compared to existing state-of-the-art models. The source code for this work is available at* https://github.com/hby123123/CrackSSM.

1. Introduction

In real-world scenarios, objects frequently develop cracks of varying degrees due to anthropogenic or environmental factors. For instance, asphalt pavements subjected to alternating temperatures and heavy loads are prone to numerous crack types. Developing efficient crack segmentation (CS) methods is therefore critical for intelligent inspection and maintenance [22]. Compared to conventional objects, cracks exhibit more complex geometries in terms of length, width, and shape, and often blend seamlessly into the background, posing substantial challenges for accurate CS.

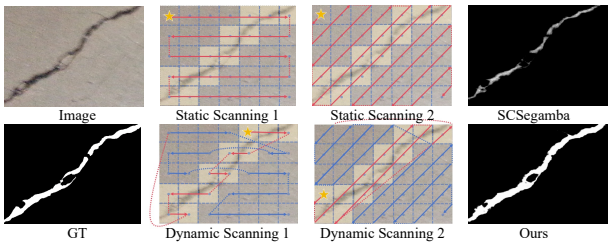


Figure 1. Static-path scanning vs. dynamic-path scanning.

To cope with these challenges, various segmentation models [4, 5] have been explored. Early efforts based on convolutional neural networks (CNNs) [2, 30] focused on local patterns and achieved reasonable results in structured scenes. However, their limited receptive fields made it difficult to model the global continuity of cracks. Transformer-based architectures [1, 10], enabled by self-attention [24], improved long-range contextual reasoning but introduced quadratic complexity, limiting their scalability in high-resolution or real-time settings.

In response to the need for both efficiency and expressiveness, state space models (SSMs) have recently emerged as a promising alternative [3, 7, 13]. Among them, Mamba [3], built on the Selective State Space (S6) mechanism, achieves efficient long-range dependency modeling with linear time complexity. This makes it particularly attractive for dense prediction tasks such as semantic segmentation. To extend Mamba to visual domains, VMamba [21] proposes a multi-directional scanning strategy that flattens 2D feature maps into 1D sequences along several spatial

axes. This design attempts to preserve more spatial context than unidirectional SSMs, and demonstrates competitive results on general-purpose vision tasks.

Building on this idea, recent efforts have applied similar strategies to CS. For example, SCSegamba [16] introduces a diagonal snake-like scanning path and integrates SSMs into a lightweight encoder for crack analysis. However, both VMamba and SCSegamba rely on static scanning paths predefined across all images. While multi-directional scanning improves over purely unidirectional designs, it still treats all images uniformly, without adapting to specific spatial structures. When applied to crack scenes, such rigid flattening further disrupts spatial continuity—pixels that are adjacent in 2D space may become distant in the 1D sequence, especially in the case of curved or disjointed cracks, as shown in Fig. 1. This spatial inconsistency weakens the causal modeling capacity of S6, which relies on temporal or sequential coherence to propagate information effectively. As a result, these methods fall short when modeling the irregular structures inherent to cracks.

In this paper, we propose **CrackSSM**, which revives the performance of S6 on crack segmentation by designing an adaptive token reordering strategy. Rather than applying a fixed scan order, we compute directional response intensities from high-level features to assess how strongly cracks are expressed along four principal orientations—horizontal, vertical, and the two diagonals. These directional cues guide the reordering of the token sequence, ensuring that semantically related and spatially connected crack regions remain adjacent in the 1D representation. By aligning the scanning path with the intrinsic structure of cracks, the model enables S6 to better capture causal dependencies across fragmented patterns while preserving its linear efficiency. Compared to the latest SOTA work-SCSegamba, which relies on a fixed diagonal scanning path, our method offers better adaptability to irregular crack patterns, making it more suitable for practical deployment.

While the encoder benefits from this spatially coherent sequence modeling, preserving boundary detail in the decoder remains equally important. Conventional upsampling often blurs fine crack edges, especially in thin or low-contrast regions. To enhance boundary fidelity, we further design two wavelet-guided decoding modules that incorporate high-frequency structural cues extracted from the original image via Haar wavelet transform. The wavelet-based gating module (WGM) generates selective gating weights to refine features at each decoding stage, while the global re-optimization module (GRO) computes pixel-wise similarity with the high-frequency map to suppress noise and enhance object-related detail expression. Extensive experiments on three benchmark datasets demonstrate that our framework achieves superior accuracy while maintaining a compact and efficient architecture suitable for real-world deployment. In summary, our main contributions are as follows:

- We propose a novel adaptive token reordering algorithm that adopts multi-direction crack feature intensities as key to reorder a static scan-derived sequence, thereby reinforcing the S6's capacity for modeling features.
- We design a WGM and a GRO based on the high-frequency priors of the input image, effectively boosting fine-detail representation and noise suppression during the decoding stage.
- We develop a novel crack segmentation model, CrackSSM, which achieves superior segmentation accuracy with low parameters and FLOPs.

2. Related Work

2.1. State Space Model

State Space Models (SSMs) have emerged as potent tools for modeling sequential data, thanks to their ability to capture long-range dependencies and support parallel computation [3]. Adapting SSMs to vision tasks [6, 27] has led to various multi-directional scanning strategies, including raster scanning [21], serpentine scanning [16], and localized scanning [11]. A latest CS example is the Mamba-based model, SCSegamba [16]. However, most current methods rely on predefined scanning paths, potentially disrupting intrinsic spatial continuity when converting 2D image features to 1D sequences—an issue that intensifies with complex shapes. We address these shortcomings by dynamically reordering target-related tokens based on multi-directional crack feature responses, enhancing the capacity of SSM to handle complex target morphologies.

2.2. Crack Segmentation

CS is crucial for a variety of real-world structures, helping reduce maintenance costs and improve efficiency. In recent years, several approaches to CS have been proposed. Among previous CNN-based methods [14, 20], DeepCrack [20] introduced an end-to-end model for CS, while BARNet [8] fused image gradient information with deep features, thereby improving boundary representation. With the rise of Transformers, numerous attention-based models have been applied to CS [1, 10], further boosting accuracy. However, such methods often suffer from high computational complexity. More recently, SSMs [3] have attracted attention for their efficient long-range modeling capabilities. A representative example is SCSegamba [16], which adopts a unique multi-directional scanning mechanism and has emerged as the latest SOTA in CS task. Unlike SCSegamba, this work leverages SSM's parallel modeling capacity while incorporating an adaptive token reordering strategy to further strengthen feature sequence learning, thereby achieving superior performance.

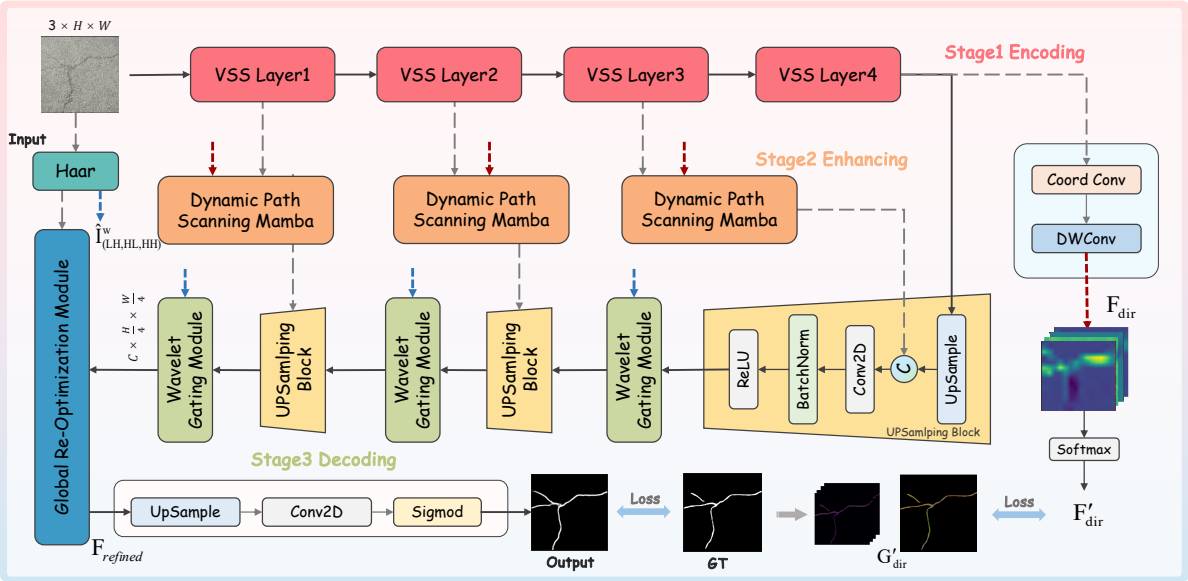


Figure 2. The overall pipeline of the proposed model, CrackSSM, which includes our designed key module, Dynamic Path Scanning Mamba module (DPSM), Wavelet Gating Module (WGM), and Global Re-Optimization Module (GRO).

3. The Proposed Model: CrackSSM

3.1. Overall Pipeline of Model

Fig. 2 depicts the overall pipeline of the proposed CrackSSM, with differently colored regions denoting three sub-networks: the *Encoding Stage*, the *Feature Enhancement Stage*, and the *Decoding Stage*. The encoding stage primarily extracts multi-scale features from the input image. Specifically, to balance efficiency in feature extraction with global perception capabilities, we adopt a vanilla VSS-based backbone [21]. This backbone comprises four feature extraction blocks, yielding multi-scale feature maps $\{\mathbf{F}_i^{enc} \in \mathbb{R}^{2^{i-1}C\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}\}_{i=1}^{4}$, where H and W indicate the height and width of the image, respectively, and C is the basic channel number. After the feature extraction, the model transitions into the *Feature Enhancement Stage*, where a novel dynamic path planning approach, shown in Fig. 3, leverages multi-directional crack intensity information to reorder an initially static path sequence. This reordering is designed to enhance the causal inference and global context modeling capacity of SSM, thereby guiding the model to focus more effectively on crack features. The resulting enhanced multi-scale encoded features $\{\hat{\mathbf{F}}_i^{enc} \in \mathbb{R}^{2^{i-1}C\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}\}_{i=1}^{4}$ advance to the decoding stage, which integrates and refines these features to better capture fine-grained details. The decoding pipeline fuses features top-down; each stage contains a simple upsampling block and a wavelet gating module (WGM), and the processed output is passed to the next stage. Unlike vanilla multi-scale feature decoding strategies, the proposed approach additionally capitalizes on high-frequency priors extracted from the original image's wavelet sub-bands, thereby alleviating smoothing effects introduced by up/-downsampling. Before obtaining the final output, a global re-optimization module (GRO) is introduced to further enrich details and suppress noise. For brevity, the following subsections concentrate on the two critical stages-feature enhancement and decoding-and their associated modules.

3.2. Adaptive Token Reordering for Structural-Aware Modeling

Standard Mamba-based vision models flatten 2D feature maps into 1D sequences using fixed scanning paths (e.g., horizontal, diagonal), which often disrupt spatial continuity—especially for thin, fragmented cracks. To address this, we propose a novel Dynamic Path Scanning Mamba (DPSM) module (as shown in Fig. 3), the core of which is an adaptive token reordering (ATR) strategy, and importantly, it requires no modification to the original S6 structure of Mamba. The ATR follows the content-aware mechanism that reorders tokens so that spatially and semantically related crack regions remain adjacent in the sequence. Our approach operates on the highest-level feature map $\mathbf{F}_4^{enc} \in \mathbb{R}^{8C\times\frac{H}{32}\times\frac{W}{32}}$, which contains rich semantic information about crack topology. We first incorporate positional information using a lightweight coordinate convolution [19], reducing the feature to four channels corresponding to horizontal, vertical, and two diagonal orientations, forming the feature map $\mathbf{F}_{dir} \in \mathbb{R}^{4\times\frac{H}{32}\times\frac{W}{32}}$. To ensure $\mathbf{F}_{dir}$ learns explicit directional sensitivity rather than generic semantics, we supervise it with a direction-aware auxiliary

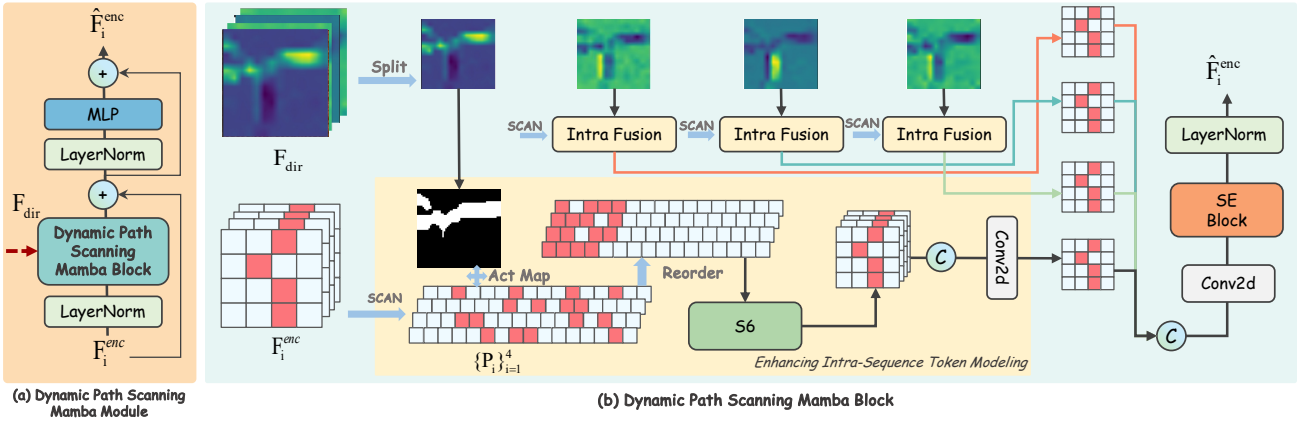


Figure 3. The workflow of the Dynamic Path Scanning Mamba module (DPSM), featuring the adaptive tokens reordering strategy. Directional intensities derived from high-level semantics guide the reordering of statically scanned sequences, ensuring spatially coherent crack regions remain adjacent in the 1D representation.

task. Specifically, we convolve the ground-truth segmentation mask with four oriented Sobel-like kernels, producing a four-channel directional label map $\mathbf{G}_{dir}$. After applying softmax across channels, we obtain the target probability distribution $\mathbf{G}'_{dir}$:

$$\mathbf{G}'_{dir}(i,j,c) = \frac{\exp\big(\mathbf{G}_{dir}(i,j,c)\big)}{\sum_{k=1}^{4}\exp\big(\mathbf{G}_{dir}(i,j,k)\big)}, \quad c \in \{1,2,3,4\}, \tag{1}$$

where (i,j) denotes the spatial coordinate and c indexes the channel. Similarly, we normalize $\mathbf{F}_{dir}$ into $\mathbf{F}'_{dir}$ via channel-wise softmax. Background pixels are explicitly masked by setting their probability vectors to $[0,0,0,0]$, ensuring supervision focuses only on foreground crack regions. The discrepancy between $\mathbf{G}'_{dir}$ and $\mathbf{F}'_{dir}$ is minimized through a cross-entropy loss, encouraging the model to predict dominant crack orientations at each location.

Equipped with $\mathbf{F}_{dir}$ as directional guidance, we perform adaptive tokens reordering on lower-level features $\{\mathbf{F}_i^{enc}\}_{i=1}^{3}$. Specifically, for each scale, we first generate four initial 1D sequences $\{\mathcal{P}_i\}_{i=1}^{4}$ by scanning the 2D feature map along four predefined snake-like paths corresponding to the same orientations. Each sequence is then reordered based on the intensity values in the upsampled $\mathbf{F}_{dir}$, such that tokens associated with strong directional responses—indicative of salient crack structures—are brought closer together in the sequence:

$$\hat{\mathcal{P}}_k = \mathrm{Sort}(\mathcal{P}_k,\ \mathrm{key} = \mathbf{F}_{dir}), \quad k = 1,2,3,4. \tag{2}$$

These reordered sequences $\{\hat{\mathcal{P}}_i\}_{i=1}^{4}$ are independently processed by standard S6 blocks of Mamba [3] to model token dependencies inside each sequence, obtainging $\{\bar{\mathcal{P}}_k\}_{i=1}^{4}$:

$$\bar{\mathcal{P}}_k = \mathrm{S6}(\hat{\mathcal{P}}_k). \tag{3}$$

Then, the resulting enhanced sequences are then mapped back to their original spatial positions to reconstruct 2D feature maps $\{\mathbf{F}_{pi}\}_{k=1}^{4}$, which are concatenated along the channel dimension and fused via a point-wise convolution:

$$\mathbf{F}^{i}_{\mathrm{merge}} = \mathrm{PointConv}\big(\mathrm{Concat}(\mathbf{F}_{p1},\ \mathbf{F}_{p2},\ \mathbf{F}_{p3},\ \mathbf{F}_{p4})\big). \tag{4}$$

We repeat the above process for each channel of $\mathbf{F}_{dir}$, generating four augmented feature maps $\{\mathbf{F}^{i}_{\mathrm{merge}}\}_{i=1}^{4}$. Then a SE block [9] is adopted to further refines these feature channels through adaptive weighting, followed by a residual structure to produce the final enhanced features $\{\hat{\mathbf{F}}_i^{enc} \in \mathbb{R}^{2^{i-1}C\times\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}}\}_{i=1}^{3}$.

Except for $\mathbf{F}_4^{enc}$, other encoded features are enhanced and modeled through the aforementioned steps. Through this process, the S6's modeling of multi-scale encoded feature maps is further optimized, enabling the model to concentrate on learning crack features.

3.3. Refining Detail Representation in Decoding

The enhanced multi-scale features $\{\hat{\mathbf{F}}_i^{enc}\}_{i=1}^{4}$ proceed to the decoding stage. To mitigate the degradation of fine crack structures caused by smoothing and loss of target details during repeated sampling and interpolation, we propose a wavelet-guided decoding strategy. This strategy enhances segmentation refinement while suppressing background noise via two complementary modules: the Wavelet Gating Module (WGM) and the Global Re-optimization Module (GRO) (Fig. 4). Both modules leverage the high-frequency sub-bands derived from the Haar wavelet transform of the original image, operating at intermediate and final decoding stages, respectively.

Specifically, we begin by decomposing the input image $\mathbf{I}_{in} \in \mathbb{R}^{3\times H\times W}$ using a single-level Haar wavelet transform:

$$\{\hat{\mathbf{I}}^{w}_{LL}, \hat{\mathbf{I}}^{w}_{LH}, \hat{\mathbf{I}}^{w}_{HL}, \hat{\mathbf{I}}^{w}_{HH}\} = \mathrm{HaarTransform}(\mathbf{I}_{\mathrm{in}}), \tag{5}$$

yielding one low-frequency component ($\hat{\mathbf{I}}^w_{LL}$) and three high-frequency sub-bands ($\hat{\mathbf{I}}^w_{LH}$, $\hat{\mathbf{I}}^w_{HL}$, $\hat{\mathbf{I}}^w_{HH}$) that capture edge-like details in horizontal, vertical, and diagonal orientations. These high-frequency components serve as explicit structural priors reflecting local intensity variations. The WGM leverages high-frequency priors to perform multi-scale segmentation refinement. At each decoding stage i, the three high-frequency maps are independently processed through a lightweight network consisting of two 1×1 convolutional layers with BatchNorm and ReLU activation. The output of this pathway passes through a sigmoid function to produce spatial gating weights:

$$\mathbf{W}_* = \sigma\big(\mathrm{Conv}_{1\times1}\big(\mathrm{ReLU}(\mathrm{BN}(\mathrm{Conv}_{1\times1}(\hat{\mathbf{I}}^w_*)))\big)\big) \quad (6)$$

where $\sigma(\cdot)$ ensures values lie in $[0, 1]$, enabling soft modulation of feature responses. Each gating map is resized to match the current feature resolution and applied to modulate the up-sampled feature map $\mathbf{F}^{dec}_i$:

$$\mathbf{F}^*_{\mathcal{G}} = \mathrm{Resize}(\mathbf{W}_*) \odot \mathbf{F}^{dec}_i. \quad (7)$$

This operation selectively amplifies features aligned with strong image gradients—such as crack boundaries—while attenuating regions lacking high-frequency support. Further, the modulated outputs from all three directions are then concatenated with the original $\mathbf{F}^{dec}_i$ and fused:

$$\mathbf{F}_{\mathrm{out}} = \mathrm{Conv}_{1\times1}\big(\mathrm{Concat}(\mathbf{F}^{dec}_i, \mathbf{F}^{LH}_{\mathcal{G}}, \mathbf{F}^{HL}_{\mathcal{G}}, \mathbf{F}^{HH}_{\mathcal{G}})\big). \quad (8)$$

By injecting wavelet-derived structural knowledge at every decoding level, WGM enables continuous guidance from the original image domain, significantly improving the model's ability to recover thin, fragmented cracks without introducing extra parameters.

Despite this local enhancement, residual noise may still persist, particularly in complex scenes. To achieve holistic refinement, we introduce the GRO at the final decoding stage, which performs global noise suppression through adaptive feature recalibration. GRO first aggregates the three high-frequency sub-bands into a unified structural map:

$$\mathbf{I}_{hf} = \frac{1}{3}\left(\hat{\mathbf{I}}^w_{LH} + \hat{\mathbf{I}}^w_{HL} + \hat{\mathbf{I}}^w_{HH}\right) \in \mathbb{R}^{3\times H\times W}. \quad (9)$$

It then projects both the final decoded feature $\mathbf{F}^3_{out}$ and $\mathbf{I}_{hf}$ into a shared latent space using linear transformations followed by ReLU activation:

$$\mathbf{X} = \mathrm{ReLU}\big(\mathrm{Linear}(\mathbf{F}^3_{out})\big), \quad \mathbf{Y} = \mathrm{ReLU}\big(\mathrm{Linear}(\mathbf{I}_{hf})\big), \quad (10)$$

ensuring semantic compatibility before comparison. A pixel-wise similarity map is computed via element-wise multiplication and a learnable projection:

$$\mathbf{S} = \sigma\big(\mathrm{Proj}(\mathbf{X} \odot \mathbf{Y})\big), \quad (11)$$

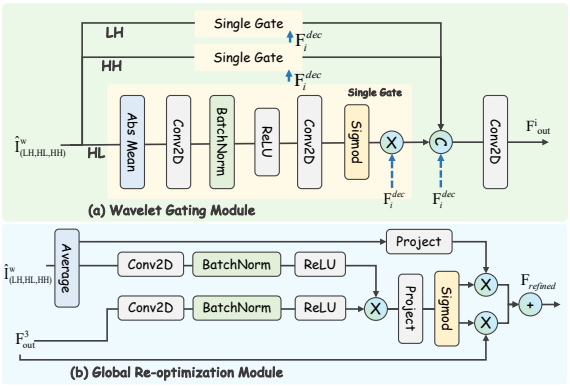


Figure 4. The detailed structure of WGM and GRO.

where $\mathrm{Proj}(\cdot)$ maps the fused representation back to the original spatial dimension, and $\sigma(\cdot)$ produces a confidence score in $[0, 1]$ for each pixel. This similarity measure reflects how well the decoded features align with true image edges. Finally, GRO adaptively fuses the decoded features with the projected high-frequency prior:

$$\mathbf{F}_{\mathrm{refined}} = (\mathbf{1} - \mathbf{S}) \odot \mathbf{F}^3_{out} + \mathbf{S} \odot \mathrm{Proj}(\mathbf{I}_{hf}), \quad (12)$$

where $\mathbf{1}$ denotes an all-ones tensor of matching size. In regions with high similarity (e.g., along real cracks), $\mathbf{S}$ approaches 1, reinforcing edge-aware details; in inconsistent areas, $\mathbf{S}$ remains low, allowing the model to suppress spurious responses. This mechanism acts as a global consistency checker, enhancing prediction reliability.

After GRO processing, the refined feature map undergoes a final upsampling and channel reduction to generate the binary crack segmentation result. Together, WGM and GRO form a hierarchical refinement decoding procedure.

3.4. Optimized Objective

We employ a multi-task learning strategy to optimize the model parameters. In the main segmentation task, we use Binary Cross Entropy (BCE) and Dice losses to supervise the alignment between the predicted maps (PM) and the ground-truth (GT). Furthermore, as described in the Static Sequence Reordering section, we apply filtering along four principal directions to the GT to derive a four-channel response map. After a softmax operation, this map captures the pixel's extension probabilities across the four directions, and background pixels are assigned a probability vector of $[0, 0, 0, 0]$ to differentiate them from foreground regions. For the direction-specific features, we adopt a Cross-Entropy (CE) loss to measure the discrepancy between the extracted directional feature distribution and the true directional response for target pixels. Consequently, the overall loss function is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{main} + \alpha \cdot \mathcal{L}_{dir}, \quad (13)$$

Table 1. Performance comparison of different methods on three crack segmentation datasets.

Methods	Crack500 [29]						TUT [15]						DeepCrack [20]					
	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑
CTCrackSeg$_{23}$	0.694	0.706	0.694	0.775	0.732	0.759	0.794	0.800	0.820	0.820	0.820	0.830	0.882	0.890	0.901	0.890	0.895	0.893
DTrCNet$_{23}$	0.701	0.724	0.653	**0.828**	0.736	0.763	0.799	0.807	0.797	0.844	0.820	0.833	0.847	0.851	0.891	0.825	0.857	0.866
Crackmer$_{24}$	0.693	0.710	0.699	0.757	0.727	0.759	0.743	0.764	0.750	0.766	0.758	0.797	0.871	0.879	0.895	0.878	0.886	0.884
SimCrack$_{24}$	0.713	0.731	0.709	0.798	0.752	0.772	0.798	0.809	0.805	0.837	0.821	0.833	0.857	0.872	0.898	0.855	0.876	0.874
CSMamba$_{24}$	0.693	0.716	0.686	0.782	0.732	0.759	0.788	0.795	0.795	0.835	0.815	0.826	0.874	0.877	0.903	0.886	0.894	0.886
MambaIR$_{24}$	0.704	0.719	0.723	0.768	0.744	0.766	0.786	0.793	0.788	0.839	0.812	0.825	0.880	0.884	0.906	0.890	0.898	0.891
PlainMamba$_{24}$	0.704	0.717	0.717	0.756	0.736	0.768	0.787	0.797	0.770	0.856	0.810	0.825	0.865	0.867	0.905	0.866	0.885	0.879
SCSegamba$_{25}$	0.714	0.733	0.697	0.805	0.746	0.771	0.805	0.812	**0.827**	0.821	0.824	0.838	0.886	0.893	0.898	0.907	0.903	0.896
DefMamba$_{25}$	0.711	0.745	0.723	0.783	0.752	0.772	0.802	0.819	0.826	0.836	0.823	0.836	0.895	0.907	0.905	0.908	0.907	0.900
CrackSSM (Ours)	**0.726**	**0.760**	**0.727**	0.805	**0.764**	**0.780**	**0.826**	**0.838**	**0.827**	**0.864**	**0.845**	**0.851**	**0.902**	**0.914**	**0.918**	**0.918**	**0.918**	**0.909**

$$\mathcal{L}_{main} = \beta \cdot \mathcal{L}_{BCE}(\mathrm{PM}, \mathrm{GT}) + \mathcal{L}_{Dice}(\mathrm{PM}, \mathrm{GT}), \quad (14)$$

$$\mathcal{L}_{dir} = \mathcal{L}_{CE}(\mathrm{G}'_{\mathrm{dir}}, \mathrm{F}_{\mathrm{dir}}{}'). \quad (15)$$

In our experimental setting, α and β are set to 0.5 and 3, respectively.

4. Experiment and Analysis

4.1. Datasets and Implementation Details

We validate the proposed method on three public benchmark datasets (Crack500 [29], DeepCrack [20], and TUT [15]), which encompass cracks or scratches on roads, walls, and other surfaces under diverse scenarios. During training, we resize all input images to 448×448 and apply classic data augmentations to improve generalization. We use AdamW with a batch size of 16, an initial learning rate of 8×10^{-4}, and a progressive decay schedule, training for up to 300 epochs and stopping early upon convergence. For evaluation, we employ widely used evaluation metrics, including Precision (P), Recall (R), F1 score, mIoU, Optimal Data Scale (ODS), and Optimal Image Scale (OIS). ODS assesses model accuracy under a global threshold, whereas OIS uses per-image thresholds to determine an upper bound in ideal settings. Beyond segmentation accuracy, we also evaluate model size, computational overhead. All experiments are conducted on NVIDIA 3090 GPUs.

4.2. Comparisons with state-of-the-arts

In this section, we compare our approach with several representative SOTA arts in crack detection—namely CTCrackSeg [23], DTrCNet [26], SimCrack [12], Crackmer [25], and SCSegamba [16]—as well as advanced methods in other computer vision tasks, such as CSMamba [18], PlainMamba [28], MambaIR [7], and dynamica scanning based model-DefMamba [17].

4.2.1. Quantitative Results and Analysis

Table 1 reports the crack segmentation results on three benchmarks, where higher values indicate better performance. From Table 1, it is evident that our proposed CrackSSM method is consistently at or near the top across all evaluation indices. In particular, for the Crack500 and DeepCrack datasets—which provide relatively fewer training data—CrackSSM achieves leading performance. On the more complex TUT dataset, CrackSSM demonstrates a larger performance margin compared to other SOTA methods, indicating that both our enhanced feature modeling strategy and optimized decoding mechanism substantially benefit segmentation accuracy. Furthermore, Table 2 compares the parameter count, computational cost, and model size of each method. Although our proposed model does not have the smallest parameter count, it maintains a relatively low computational cost while still achieving the best segmentation results.

Table 2. Comparison of FLOPS, parameter number, and memory usage among different methods.

Methods	FLOPS↓	Params↓	Size↓
CTCrackSeg$_{23}$	39.47G	22.88M	174MB
CSMamba$_{23}$	145.84G	35.95M	233MB
DTrCNet$_{24}$	39.47G	22.88M	174MB
Crackmer$_{24}$	14.94G	5.90M	43MB
SimCrack$_{24}$	286.62G	29.58M	225MB
MambaIR$_{24}$	47.32G	10.34M	79MB
PlainMamba$_{24}$	73.36G	16.72M	96MB
DefMamba$_{25}$	50.91G	177.44M	201MB
SCSegamba$_{25}$	18.16G	**2.80M**	37MB
CrackSSM(Ours)	**4.69G**	2.95M	**15MB**

4.2.2. Qualitative Results and Analysis

To further illustrate each method's performance in complex scenarios, we present visual comparisons in Fig. 5. The results show that CrackSSM excels in terms of segmentation integrity, accuracy, and background noise suppression, highlighting the strong generalization capabilities of our approach across diverse environments.

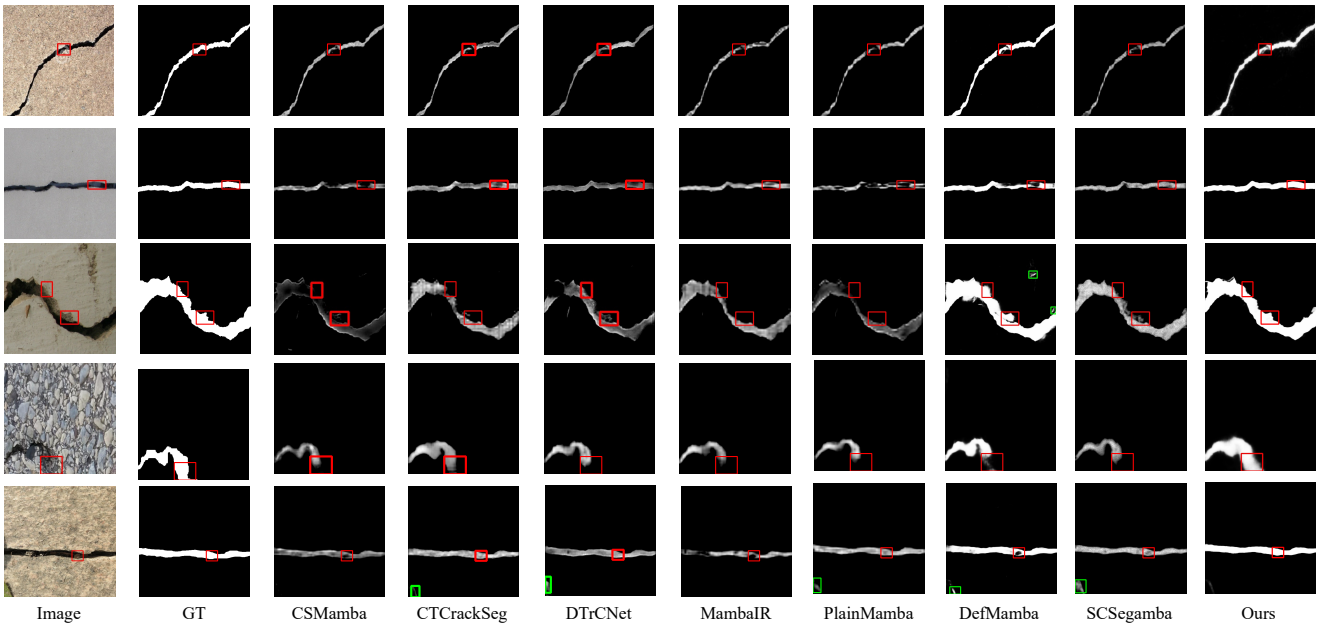


Figure 5. Qualitative comparison results of recently proposed SOTA methods. Red boxes highlight the detailed differences in detection, while green boxes mark the false positives.

Table 3. Performance on the TUT dataset under different module configurations. "✓" indicates that the module is enabled; "×" indicates that it is disabled.

DPSM	WGM	GRO	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑
×	×	×	0.791	0.804	0.799	0.824	0.811	0.827
✓	×	×	0.803	0.816	0.809	0.839	0.824	0.836
×	✓	×	0.805	0.818	0.810	0.844	0.827	0.837
×	×	✓	0.802	0.814	0.805	0.845	0.825	0.835
✓	✓	×	0.807	0.821	0.809	0.852	0.830	0.839
✓	×	✓	0.807	0.819	0.813	0.845	0.829	0.839
×	✓	✓	0.807	0.821	0.811	0.847	0.829	0.839
✓	✓	✓	0.826	0.838	0.827	0.864	0.845	0.851

4.3. Ablation Studies

4.3.1. Impact of Key Components

We conducted a module ablation study to systematically evaluate the contribution and performance gain of each components within the CrackSSM. We define the baseline setting: the encoder utilizes the VSSM backbone to extract multi-level features, which are directly fed to the corresponding decoder layers for upsampling and feature fusion, thereby generating results. In addition, we evaluated the model's performance under seven distinct configurations, as shown in Table 3. When different combinations of DPSM, WGM, and GRO are added to the baseline, the model's performance improves to varying degrees. When only a single module is enabled, each metric increases significantly (see Rows 2–4), indicating that any of the direction-awareness, detail enhancement, or global structure re-optimization modules can effectively remedy the shortcomings of the baseline model. Furthermore, pairing any two modules offers complementary advantages in direction capture, detail restoration, and structural consistency, leading to better performance than single-module configurations (see Rows 5–7). Notably, when all three modules are integrated, the model synergistically addresses high-level feature direction sensing, local detail enhancement, and global noise suppression; all metrics reach their optimal values, confirming the superior performance of the complete model in complex crack scenarios. Besides, Fig. 6 provides a detailed qualitative visualization of our proposed WGM and GRO. The series of "WGM Before/After" pairs demonstrates a progressive feature refinement process. From the optimization process of feature content, we can see that WGM iteratively enhance activations along the crack paths while suppressing irrelevant background responses, making the crack features more salient and continuous. Finally, the "GRO Before/After" comparison highlights its role in global output refinement. GRO effectively eliminates isolated noise artifacts and enhances the connectivity of the crack network, resulting in a more reliable output.

4.3.2. Feature's Direction-aware Learning

In the Methods section, we mentioned that four direction-aware labels were introduced in the DPSM to supervise the learning of higher-level features, thereby uncovering directional information beyond semantic features. To verify whether this kind of direction-aware learning can further enhance model performance, we conducted experiments with

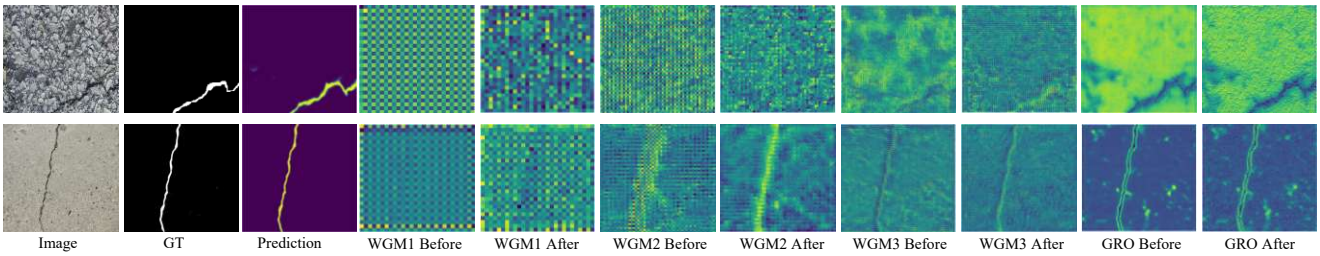


Figure 6. Visualization of the effectiveness of WGM and GRO modules.

Table 4. The results of w/ vs. w/o direction-aware loss ($\mathcal{L}_{dir}$).

$\mathcal{L}_{dir}$	Dataset	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑
w/o	DeepCrack	0.893	0.908	0.911	0.908	0.911	0.902
	TUT	0.816	0.829	0.824	0.853	0.838	0.845
	Crack500	0.712	0.748	0.715	0.790	0.750	0.771
w/	DeepCrack	0.902	0.914	0.918	0.918	0.918	0.909
	TUT	0.826	0.838	0.827	0.864	0.845	0.851
	Crack500	0.726	0.760	0.727	0.805	0.764	0.780

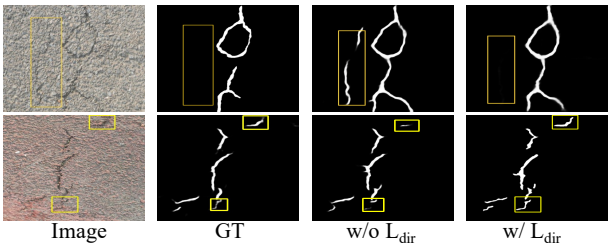


Figure 7. The results of w/o and w/ direction-aware learning.

and without direction-aware loss function on three datasets. The outcomes is shown in Table 4. It can be observed that, although the model already achieves relatively favorable segmentation metrics (compared to other methods) using the unconstrained approach, adding explicit supervision via direction-sensitive labels improves the metrics to varying degrees across all datasets. For instance, on the DeepCrack dataset, ODS increases from 0.893 to 0.902, and F1 increases from 0.911 to 0.917; similar enhancements can also be observed on the other datasets. To provide an intuitive comparison of performance under both settings, Fig. 7 shows visualization results in a multi-branching crack scenario, where our method demonstrates superior detail depiction and overall coherence. These experiments confirm that employing a adaptive tokens reordering strategy based on direction sensitivity effectively strengthens S6's ability to model crack characteristics, further validating the effectiveness of our proposed approach.

Furthermore, we also explore the role of supervision labels in the direction-aware mechanism, we apply different label configurations to $\mathbf{F}_4^{enc}$: (1) Two labels capturing horizontal and vertical direction changes; (2) Two labels capturing only the diagonal directions; (3) The four complete directions as used in our main text. The results of different configurations are listed in Table. 5. As the results show, using labels in four basic directions for supervised learning leads to better model performance, further confirming the rationality of our design.

Table 5. The results adopting various direction labels in DPSM.

Direction Label	ODS↑	OIS↑	P↑	R↑	F1↑	mIoU↑
Horizontal and Vertical	0.819	0.832	0.820	0.858	0.838	0.847
Two Diagonals	0.816	0.830	0.821	0.857	0.839	0.845
Four Directions (Ours)	0.826	0.838	0.827	0.864	0.845	0.851

5. Conclusion

In this paper, we propose a new crack segmentation (CS) model, CrackSSM, integrating a dynamic path scanning strategy into Mamba's S6 to better handle the complex geometries of cracks while maintaining efficiency. By leveraging four principal directions in high-level crack features, the reordering of token sequences addresses the limitations of static-path scanning and effectively reduces distortion in visual spatial structure. The incorporation of a wavelet-based gating module (WGM) and a global re-optimization module (GRO) further enhances fine-grained detail modeling and noise suppression in the decoding stage. Experimental evaluations on multiple benchmark datasets confirm that our approach outperforms current state-of-the-art methods—not only in segmentation accuracy but also in reduced parameter count and computational overhead. Overall, these results demonstrate that jointly strengthening SSM's capacity for causal inference with dynamic path planning and exploiting high-frequency priors for detailed expression offers a leading solution for robust and efficient CS. In future work, we will extend the dynamic scanning strategy to other structural analysis tasks and other vision applications.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2023YFB4502804), the National Science Fund for Distinguished Young Scholars (No. 62025603), the National Natural Science Foundation of China (No. U22B2051, No. 62302411), and China Postdoctoral Science Foundation (No. 2023M732948).

References

[1] Po-Hao Chen, Jun-Wei Hsieh, Yi-Kuan Hsieh, Chuan-Wang Chang, and Deng-Yuan Huang. Cross-scale overlapping patch-based attention network for road crack detection. *IEEE Trans. Intell. Transp. Syst.*, 26(6):7587–7599, 2025. 1, 2

[2] Xu Cheng, Tian He, Fan Shi, Meng Zhao, Xiufeng Liu, and Shengyong Chen. Selective feature fusion and irregular-aware network for pavement crack detection. *IEEE Trans. Intell. Transp. Syst.*, 25(5):3445–3456, 2023. 1

[3] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First conference on language modeling*, 2024. 1, 2, 4

[4] Yubin Gu, Honghui Xu, Yueqian Quan, Wanjun Chen, and Jianwei Zheng. Orsi salient object detection via bidimensional attention and full-stage semantic guidance. *IEEE Trans. Geosci. Remote Sens.*, 61:1–13, 2023. 1

[5] Yubin Gu, Siting Chen, Xiaoshuai Sun, Jiayi Ji, Yiyi Zhou, and Rongrong Ji. Optical remote sensing image salient object detection via bidirectional cross-attention and attention restoration. *Pattern Recognit.*, 164:111478, 2025. 1

[6] Yubin Gu, Yuan Meng, Jiayi Ji, and Xiaoshuai Sun. Acl: Activating capability of linear attention for image restoration. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 17913–17923, 2025. 2

[7] Hang Guo, Jinmin Li, Tao Dai, Zhihao Ouyang, Xudong Ren, and Shu-Tao Xia. Mambair: A simple baseline for image restoration with state-space model. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 222–241. Springer, 2024. 1, 6

[8] Jing-Ming Guo, Herleeyandi Markoni, and Jiann-Der Lee. Barnet: Boundary aware refinement network for crack detection. *IEEE Trans. Intell. Transp. Syst.*, 23(7):7343–7358, 2021. 2

[9] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 7132–7141, 2018. 4

[10] Xiangkun Hu, Hua Li, Yixiong Feng, Songrong Qian, Jian Li, and Shaobo Li. Ccdformer: A dual-backbone complex crack detection network with transformer. *Pattern Recognit.*, 161:111251, 2025. 1, 2

[11] Tao Huang, Xiaohuan Pei, Shan You, Fei Wang, Chen Qian, and Chang Xu. Localmamba: Visual state space model with windowed selective scan. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 12–22. Springer, 2024. 2

[12] Achref Jaziri, Martin Mundt, Andres Fernandez, and Visvanathan Ramesh. Designing a hybrid neural system to learn real-world crack segmentation from fractal-based simulation. In *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pages 8636–8646, 2024. 6

[13] Boyun Li, Haiyu Zhao, Wenxin Wang, Peng Hu, Yuanbiao Gou, and Xi Peng. Mair: A locality- and continuity-preserving mamba for image restoration. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 7491–7501, 2025. 1

[14] Qinghua Lin, Wei Li, Xiangpan Zheng, Haoyi Fan, and Zuoyong Li. Deepcrackat: An effective crack segmentation framework based on learning multi-scale crack features. *Eng. Appl. Artif. Intell.*, 126:106876, 2023. 2

[15] Hui Liu, Chen Jia, Fan Shi, Xu Cheng, Mianzhao Wang, and Shengyong Chen. Staircase cascaded fusion of lightweight local pattern recognition and long-range dependencies for structural crack segmentation. *arXiv preprint arXiv:2408.12815*, 2024. 6

[16] Hui Liu, Chen Jia, Fan Shi, Xu Cheng, and Shengyong Chen. Scsegamba: Lightweight structure-aware vision mamba for crack segmentation in structures. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 29406–29416, 2025. 2, 6

[17] Leiye Liu, Miao Zhang, Jihao Yin, Tingwei Liu, Wei Ji, Yongri Piao, and Huchuan Lu. Defmamba: Deformable visual state space model. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 8838–8847, 2025. 6

[18] Mushui Liu, Jun Dan, Ziqian Lu, Yunlong Yu, Yingming Li, and Xi Li. Cm-unet: Hybrid cnn-mamba unet for remote sensing image semantic segmentation. *arXiv preprint arXiv:2405.10530*, 2024. 6

[19] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. *Adv. Neural Inf. Process. Syst. (NeruIPS)*, 31, 2018. 3

[20] 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. 2, 6

[21] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Adv. Neural Inf. Process. Syst. (NeruIPS)*, 37:103031–103063, 2024. 1, 2, 3

[22] Qinzhong Tan, Ao Li, Le Dong, Weisheng Dong, Xin Li, and Guangming Shi. Cds-net: contextual difference sensitivity network for pixel-wise road crack detection. *IEEE Trans. Circuits Syst. Video Technol.*, 35(6):5223–5235, 2025. 1

[23] Huaqi Tao, Bingxi Liu, Jinqiang Cui, and Hong Zhang. A convolutional-transformer network for crack segmentation with boundary awareness. In *Proc. IEEE Int. Conf. Image Process. (ICIP)*, pages 86–90. IEEE, 2023. 6

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Adv. Neural Inf. Process. Syst. (NeruIPS)*, 30, 2017. 1

[25] Jin Wang, Zhigao Zeng, Pradip Kumar Sharma, Osama Alfarraj, Amr Tolba, Jianming Zhang, and Lei Wang. Dual-path network combining cnn and transformer for pavement crack segmentation. *Automation in Construction*, 158: 105217, 2024. 6

[26] Chao Xiang, Jingjing Guo, Ran Cao, and Lu Deng. A crack-segmentation algorithm fusing transformers and convolutional neural networks for complex detection scenarios. *Automation in Construction*, 152:104894, 2023. 6

[27] Xinyu Xie, Yawen Cui, Tao Tan, Xubin Zheng, and Zitong Yu. Fusionmamba: Dynamic feature enhancement for multimodal image fusion with mamba. *Visual Intelligence*, 2(1): 37, 2024. 2

[28] Chenhongyi Yang, Zehui Chen, Miguel Espinosa Minano, Linus Ericsson, Zhenyu Wang, Jiaming Liu, and Elliot J Crowley. Plainmamba: Improving non-hierarchical mamba in visual recognition. In *35th Proc. Br. Mach. Vis. Conf. (BMVC)*, 2024. 6

[29] Fan Yang, Lei Zhang, Sijia Yu, Danil Prokhorov, Xue Mei, and Haibin Ling. Feature pyramid and hierarchical boosting network for pavement crack detection. *IEEE Trans. Intell. Transp. Syst.*, 21(4):1525–1535, 2019. 6

[30] Tianjie Zhang, Donglei Wang, and Yang Lu. Ecsnet: An accelerated real-time image segmentation cnn architecture for pavement crack detection. *IEEE Trans. Intell. Transp. Syst.*, 24(12):15105–15112, 2023. 1