information

MDPI

Article

STCYOLO: Subway Tunnel Crack Detection Model with Complex Scenarios

Jia Zhang [1], Hui Li [2,3,4,*], Weidong Song [3], Jinhe Zhang [3] and Miao Shi [4]

[1] Institute of Railway Engineering, Liaoning Railway Vocational and Technical College, Jinzhou 121000, China; zhangjia19850504@gmail.com
[2] Institute of Urban Rail Transit, Liaoning Railway Vocational and Technical College, Jinzhou 121000, China
[3] School of Mapping and Geographical Science, Liaoning Technical University, Fuxin 123000, China; swd18940693586@gmail.com (W.S.); 47221055@stu.lntu.edu.cn (J.Z.)
[4] School of Civil Engineering, Liaoning University of Technology, Jinzhou 121000, China; mini05021314@gmail.com
* Correspondence: lh13464857601@gmail.com

Abstract: The detection of tunnel cracks plays a vital role in ensuring structural integrity and driving safety. However, tunnel environments present significant challenges for crack detection, such as uneven lighting and shadow occlusion, which can obscure surface features and reduce detection accuracy. To address these challenges, this paper proposes a novel crack detection network named STCYOLO. First, a dynamic snake convolution (DSConv) mechanism is introduced to adaptively adjust the shape and size of convolutional kernels, allowing them to better align with the elongated and irregular geometry of cracks, thereby enhancing performance under challenging lighting conditions. To mitigate the impact of shadow occlusion, a Shadow Occlusion-Aware Attention (SOAA) module is designed to enhance the network's ability to identify cracks hidden in shadowed regions. Additionally, a tiny crack upsampling (TCU) module is proposed, which reorganizes convolution kernels to more effectively preserve fine-grained spatial details during upsampling, thereby improving the detection of small and subtle cracks. The experimental results demonstrate that, compared to YOLOv8, our proposed method achieves a 2.85% improvement in mAP and a 3.02% increase in the F score on the crack detection dataset.

Keywords: tunnel crack detection; shadow occlusion-aware attention; tiny crack upsampling; convolutional neural network

check for updates

Academic Editor: Marco Leo

Received: 17 May 2025
Revised: 10 June 2025
Accepted: 13 June 2025
Published: 18 June 2025

Citation: Zhang, J.; Li, H.; Song, W.; Zhang, J.; Shi, M. STCYOLO: Subway Tunnel Crack Detection Model with Complex Scenarios. *Information* **2025**, *16*, 507. https://doi.org/10.3390/info16060507

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

As of 31 December 2020, the operational mileage of urban rail transit in China (excluding Hong Kong, Macau, and Taiwan) reached 7545.5 km, with subway lines accounting for approximately 76% of this total. Safe and reliable subway tunnels are an important guarantee for the safe operation of urban subways. With the vigorous development of China's urban rail transit industry, defects in subway tunnels caused by factors such as structural aging or the dynamic impact of operating vehicles have gradually become apparent, with a large number of subway tunnels moving from the construction phase into the maintenance phase.

Crack detection in subways is a crucial aspect of subway tunnel maintenance work. The formation of cracks is usually the result of a combination of factors such as groundwater levels, geological structures, and surface loads during the construction of subway tunnels. After the subway begins operations, the formation of cracks may also be exacerbated

by factors such as vehicle vibration and geological changes. Traditional crack detection methods mainly rely on manual inspection and monitoring equipment, using engineers' visual inspection and equipment monitoring to determine the size and location of and trend in cracks. However, this method has disadvantages such as high labor costs, low efficiency, and poor accuracy, often failing to meet the needs for detailed, informative, and intelligent subway tunnel maintenance.

Non-image-based tunnel crack detection methods mainly rely on other types of sensors and data acquisition technologies to detect and evaluate cracks through physical measurements or signal analysis, such as laser scanning detection [1], geological radar detection [2], ultrasonic detection [3], line scanning cameras [4], etc. These methods are especially advantageous in environments with poor lighting or complex surface conditions where traditional imaging may be ineffective.

Image processing methods primarily analyze and process images of subway tunnel cracks to extract information about the cracks. Key techniques include edge detection [5], threshold segmentation [6], and median filtering [7]. Traditional tunnel crack extraction algorithms are sensitive to environmental factors such as lighting and noise. The complex environmental conditions inside tunnels can limit the completeness and accuracy of crack extraction. With the development of machine learning technology, machine learning methods identify subway cracks by learning patterns on the image surface. These approaches primarily include methods based on features, support vector machines, artificial neural networks, and random forests. Compared to traditional methods, machine learning approaches offer advantages in automation and accuracy, significantly improving the efficiency and precision of subway crack extraction. However, machine learning can only cater to tunnel crack detection under certain specific circumstances. The emergence of new crack environments necessitates reconfiguration, failing to meet the detection requirements for subway tunnel cracks in all situations.

Currently, with the development and application of artificial intelligence technologies [8–10], the method of extracting subway tunnel cracks is shifting towards automation. At present, subway tunnel inspections mainly include two approaches: rapid extraction based on object detection and detailed extraction based on semantic segmentation.

(1) Based on object detection

Xue [11] proposed a two-stage solution based on the R-FCN architecture, initially using an improved FCN for crack classification, followed by accurate defect localization and identification in images using components such as a region proposal network, position-sensitive RoI pooling, a softmax classifier, and a bounding box regressor. Zhou [12] introduced a method that employs WGANRE data augmentation and a crop-based YOLOv4 model to address the issues of data scarcity and the balance between model size and accuracy, thereby achieving the automatic detection of tunnel lining cracks. Li [13] proposed a model for the rapid detection of tunnel cracks, employing a pyramidal anchor method in the RPN to generate anchors of various scales and aspect ratios, accommodating the unique morphology of cracks. Additionally, a defect merging algorithm was introduced to identify and consolidate adjacent cracks in the input image, determining whether they belong to the same fracture. Juan [14] introduced a tunnel crack detection network, TDD-YOLO, based on YOLOv7, utilizing the MobileViT network as the backbone. A CA module was added after the upsampling and downsampling operations of the feature pyramid network, enhancing the network's perception of features at different scales. The integration of the new TP Block module improves the network's ability to quickly and effectively extract crack features. Dai [15] proposed an improved YOLOv5 model for the detection of subway cracks. The original SPP structure was replaced with SPPF and PANet, effectively enhancing the model's target recognition capabilities. Furthermore, a coordinate attention

module was introduced, integrating crack location information into channel attention, further improving the accuracy of crack detection.

(2) Based on semantic segmentation

Ren [16] introduced CrackSegNet, a crack extraction network with an encoder–decoder structure. The encoder pathway is based on a modified VGG-16 network, using dilated convolutions to extract additional features. The decoder utilizes spatial pyramid pooling to capture representations of different subregions, followed by upsampling and concatenation to merge features containing both local and global information. Skip connections are used to link corresponding feature maps between the encoder and decoder pathways. Li [17] proposed U-CliqueNet, a crack extraction network that integrates U-net with CliqueNet. This model, building upon the foundation of U-net, substitutes ordinary convolutional layers with clique blocks, increasing the network's depth and richness. A channel attention mechanism is added during the downsampling process, enabling the network to outperform U-net in handling crack segmentation noise. Liao [18] introduced a network named LinkCrack for rapid and accurate crack detection. It utilizes ResNet-34 as the encoder and employs techniques such as additive fusion and bilinear interpolation in the decoder to reduce parameter count and increase speed. Moreover, an adjacency connection graph is introduced as a spatial constraint to model the continuity of cracks. Pixel loss and adjacency connection loss are used as loss functions to enhance the network's accuracy in detecting cracks. Zhou [19] proposed a model named TCDNet for tunnel crack detection. It enhances feature representation through MA and RFE modules and expands the receptive field to capture more global information. A multiscale feature fusion strategy preserves rich semantic information, improving crack detection accuracy. Furthermore, a weighted BCE loss function is utilized to address the issue of sample imbalance, ensuring effective model training. Zhou [20] introduced an algorithm, LC-DeepLab, suitable for the detection of complex tunnel lining cracks. It utilizes a modified MobileNetV3 as the backbone feature extractor and incorporates shallow feature fusion modules and the ECANet attention module to enhance its ability to detect crack details.

The following problems exist:

(1) The interior of tunnels may be subject to shadow effects from lighting, impacting the visibility of cracks, causing some cracks to be obscured or blurred, thereby increasing the difficulty of crack extraction.
(2) Interpolation operations during the upsampling process may lead to the loss or blurring of information about minor cracks, resulting in the inability to effectively reconstruct these small cracks.

The contributions of this paper are as follows:

(1) Given the elongated structure of cracks, this paper proposes the introduction of the dynamic snake convolution method to enhance sensitivity to crack structures, better conform to and capture these structures, and improve crack detection performance.
(2) A TCU method is proposed, which, compared to traditional upsampling methods, can more effectively retain the features of minor cracks, avoiding the loss of important information during the upsampling process.
(3) A head with SOAA is proposed, enabling it to effectively handle scenarios where cracks are obscured by shadows.

2. Methodology

The subway tunnel environment is extremely complex, with challenges such as uneven lighting, shadows, and noise that hinder accurate crack extraction. To address these issues,

this paper proposes the STCYOLO architecture, shown in Figure 1, with three key modules designed for robustness and precision.

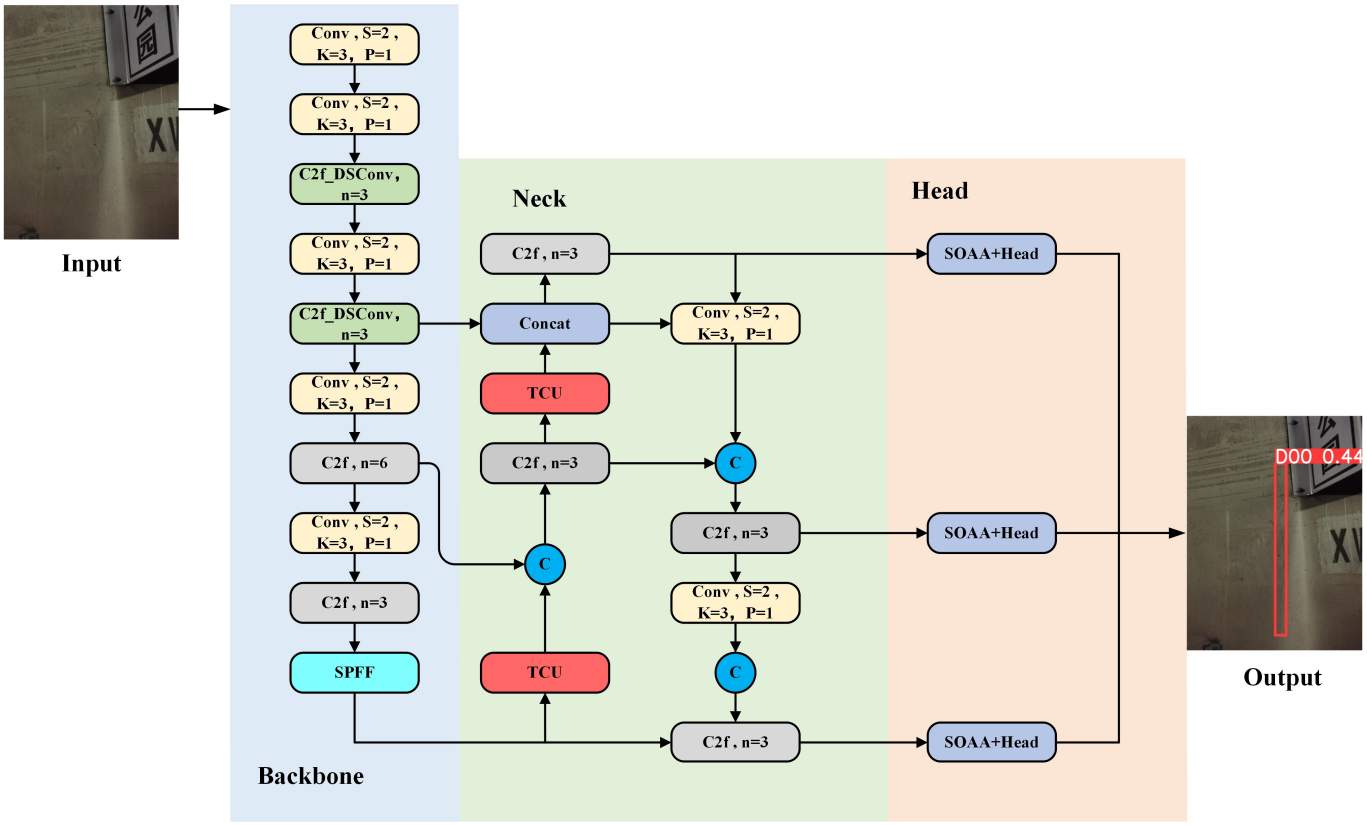


Figure 1. The structure of STCYOLO.

First, the standard convolution layers in the first two C2f blocks of YOLOv8 are replaced withDSConv, which enables the network to adapt its receptive field to elongated and irregular crack shapes, improving geometric alignment and detection sensitivity. This selective replacement avoids excessive parameters while focusing on the early feature extraction stages where structure adaptation is the most critical. Second, a TCU module is introduced to dynamically reconstruct spatial details during upsampling, helping preserve fine crack features that are often lost with traditional interpolation methods. Third, to enhance robustness under poor lighting and occlusion, the original YOLO detection head is replaced with an SOAA module. This module captures multiscale semantic cues and improves the model's focus on cracks obscured by shadows. These components form a crack detection framework that is lightweight yet highly effective in challenging subway tunnel scenarios.

2.1. Dynamic Snake Convolution

Tunnel cracks generally come in three forms: transverse cracks, longitudinal cracks, and alligator cracks. Both transverse and longitudinal cracks are slender in structure, while alligator cracks represent a collection of multiple transverse and longitudinal cracks. These cracks occupy a low proportion of the entire image. Traditional convolution operations utilize convolutional kernels of fixed size, leading to a fixed shape of the receptive field, which is clearly unreasonable for the limited pixel composition shapes such as cracks. Deformable convolutions [21] introduce learnable offsets, allowing the shape of the receptive field to adjust according to the actual shape of objects. This flexibility permits the receptive field to better adapt to the deformations of different objects. However, without constraints

or regularization on the offsets, they may become overly flexible, leading to poor capability in extracting small objects. Dynamic snake convolution [22] is established on the basis of deformable convolution by adding tubular constraints and adopting an iterative strategy (Figure 2). For each target to be processed, the following observation positions are selected sequentially, which ensures the continuity of attention and does not cause the perceptual range to be too scattered due to large deformation offset. Considering a convolution kernel of size 9 and taking the x-axis direction as an example, the specific position of each grid in K is represented as follows: $K_{i\pm c} = (x_{i\pm c}, y_{i\pm c})$, where c = {0, 1, 2, 3, 4} denotes the horizontal distance from the central grid. In Figure 3, the changes within the receptive field along the x-axis and y-axis are given by the following equation:

$$K_{i\pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_i + c, y_i + \Sigma_i^{i+c} \Delta y), \\ (x_{i-c}, y_{i-c}) = (x_i - c, y_i + \Sigma_{i-c}^{i} \Delta y), \end{cases} \tag{1}$$

$$K_{j\pm c} = \begin{cases} (x_{j+c}, y_{j+c}) = (x_j + \Sigma_j^{j+c} \Delta x, y_j + c), \\ (x_{j-c}, y_{j-c}) = (x_j + \Sigma_{j-c}^{j} \Delta x, y_j - c), \end{cases} \tag{2}$$

where K represents the fractional positions in Equations (1) and (2), and K' enumerates all integer spatial positions.

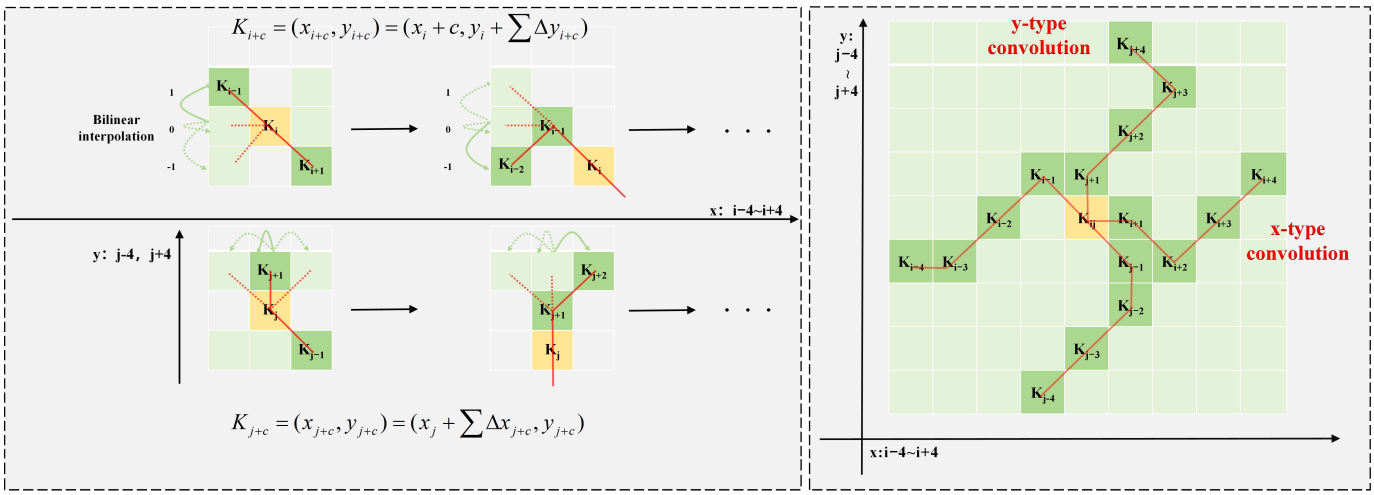


Figure 2. (**Left**): An illustration of the coordinate calculation in dynamic snake convolution (DSConv). By iteratively shifting kernel positions along the x- and y-directions with learned offsets, the receptive field adapts to elongated crack shapes. (**Right**): The resulting receptive field trajectory after applying directional constraints. This ensures spatial continuity in the perception path, enabling DSConv to effectively focus on fine, irregular crack structures without excessive dispersion.

From the heatmap in Figure 4, it can be seen that the lighting in the image is uneven, and the cracks are fine. The original YOLOV8 focuses more on the large background areas above, with almost no attention to the cracks. After incorporating C2f_DSConv, YOLOV8's focus on the cracks significantly increases, thereby proving the effectiveness of C2f_DSConv.

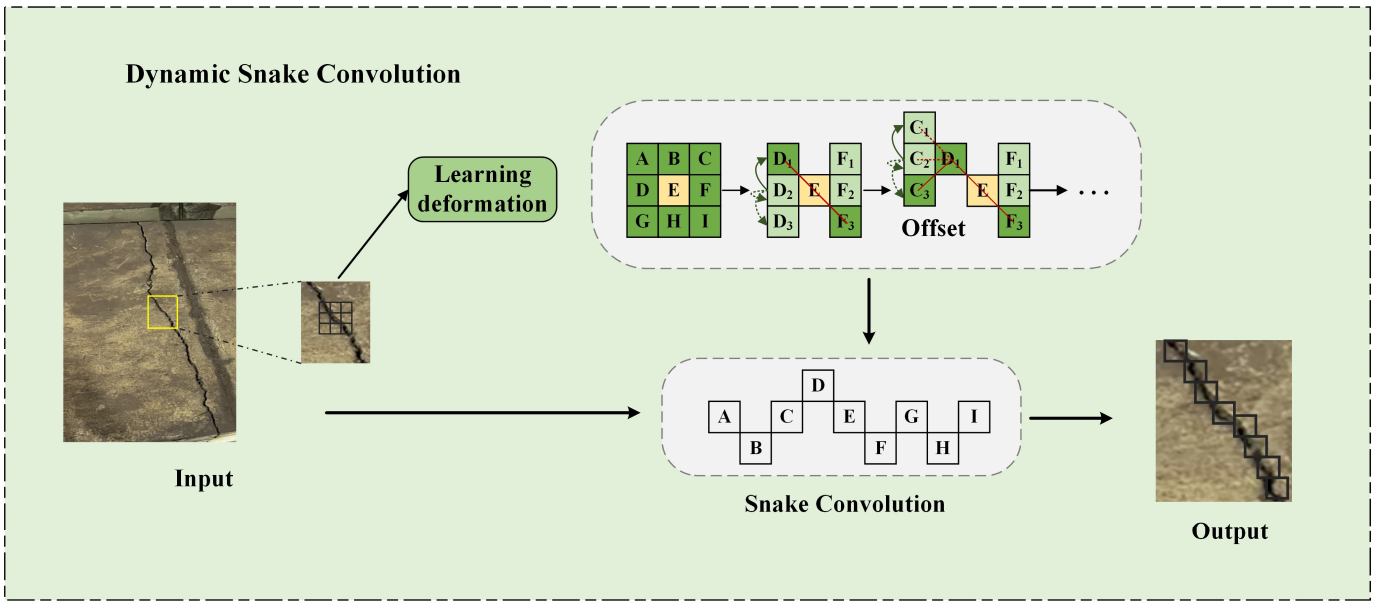


Figure 3. DSConv learns spatial offsets for each kernel position based on the shape of crack features, allowing the receptive field to adaptively follow irregular paths. After offset learning, the convolution kernel is reorganized into a snake-like path to match the geometry of cracks. This mechanism improves the model's sensitivity to elongated, fine-grained structures while maintaining spatial continuity in feature extraction.

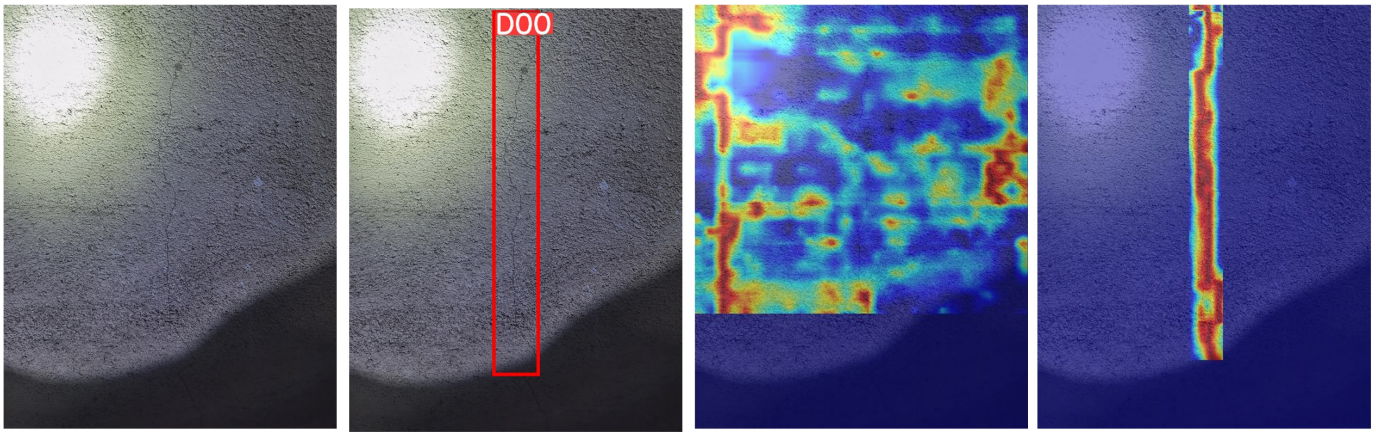


Figure 4. YOLOV8 adds dynamic snake convolution heatmap.

2.2. Tiny Crack Upsampling Algorithm

Bilinear interpolation is an interpolation method based on the values of neighboring pixels, where the value of a new pixel is estimated through the weighted average of the surrounding pixels. In this process, Subtle variations and detailed information contained in the original pixels may be lost. This is especially true for small targets such as fine cracks, which may be blurred by the average values of the surrounding pixels, leading to the neglect of crack details. Furthermore, fine objects such as cracks often exhibit spatial discontinuities, and traditional upsampling methods like bilinear interpolation may not effectively preserve these discontinuities, leading to decreased detection performance for small objects. To address the aforementioned issue, this paper introduces tiny crack upsampling, capable of flexibly adjusting and reassembling the convolutional kernel based on the different positions and contents of the input features. This allows for a better capture

of the spatial structure of and detail information on tiny crack features, achieving more precise and accurate upsampling.

Assuming the input feature map is $X \in R^{C\times H\times W}$ and setting the upsampling magnification to σ, TCU first predicts a recombination kernel based on the content of each target location and then recombines the input features based on the predicted kernel. Finally, a new feature map of size $X' \in R^{C\times \sigma H\times \sigma W}$ is generated.

As shown in Figure 5, each position on X corresponds to a target area of size σ 2 on X'. First, convolution 1×1 is used to reduce the number of channels in the feature map, reducing the complexity of subsequent calculations while preserving the main information on the feature. Next, for a convolutional layer of $k_{up} \times k_{up}$, the shape of the upsampling kernel to be predicted is $\sigma H \times \sigma W \times k_{up} \times k_{up}$.

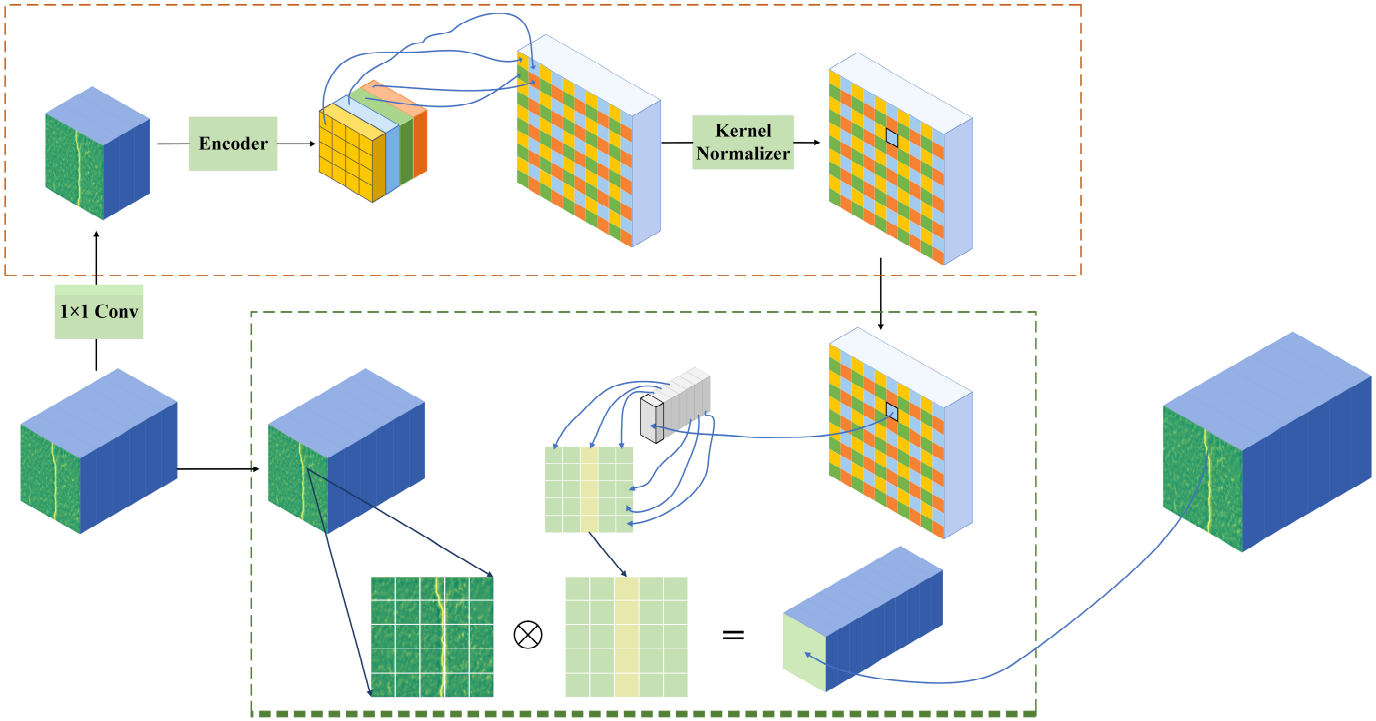


Figure 5. Structure of TCU. It uses content-aware kernel prediction mechanism to adaptively reassemble feature maps, preserving fine details of small cracks that are often lost in traditional interpolation.

For the compressed input feature map from the first step, the upsampling kernel is predicted using convolution $k_{encoder} \times k_{encoder}$ ($k_{encoder} = 3$). An output feature map with size $H \times \sigma W \times \sigma^2 k_{up}^2$,where $k_{up} = 5$ is generated, and its channel dimension in the spatial dimension is expanded to generate an upsampling kernel with a shape of $\sigma H \times \sigma W \times k_{up}^2$. Finally, the softmax function is used for normalization so that the sum of its weights is 1.

The feature recombination module first maps each position in the output feature map back to the original input feature map and extracts the region of $k_{up} \times k_{up}$ centered on it. Then, the upsampling kernel of the predicted point is used, and dot product operation is performed to obtain the final output value. During the feature recombination process, different channels at the same position will share the same upsampling kernel. The semantics of the restructured feature map are stronger than those of the original feature map, as information from relevant points in local regions can receive more attention.

From the heatmap in Figure 6, it is evident that the lighting in the image is uneven, and the cracks are fine. The original YOLOV8 pays more attention to the large background areas above, with almost no focus on the cracks. After incorporating TCU, YOLOV8's focus on the cracks significantly increases, thereby demonstrating the effectiveness of C2f_DSConv.

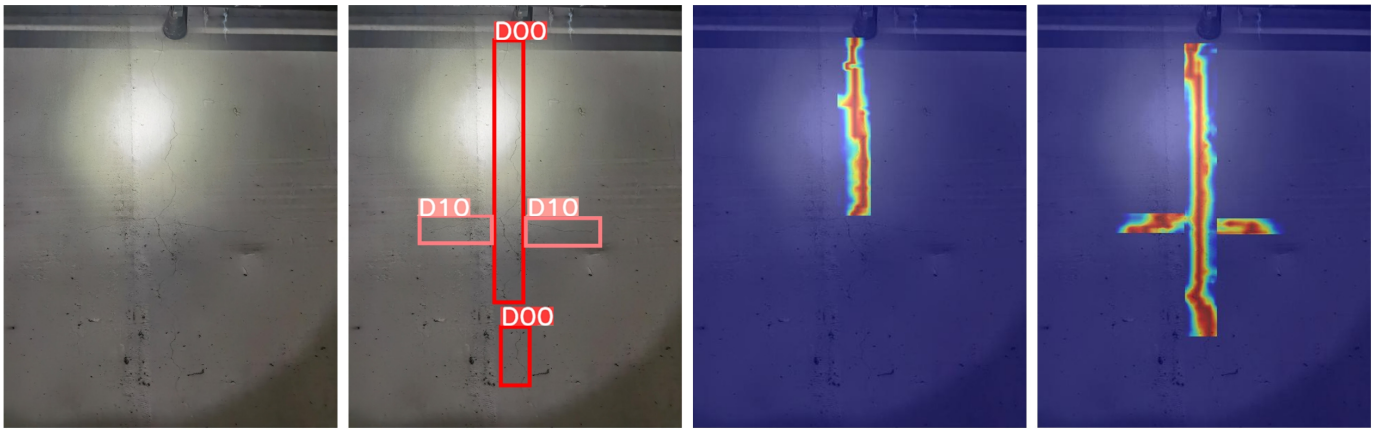


Figure 6. Heatmap of YOLOV8 with tiny crack upsampling added.

2.3. Shadow Occlusion-Aware Attention Mechanism

Tunnel environments typically suffer from dim lighting conditions that reduce image contrast and obscure crack features. Additionally, surface contaminants such as stains, dust, and debris further complicate crack detection, posing significant challenges for precise crack localization and identification. To address these issues, this paper proposes a multihead attention network named SOAA, which effectively captures multiscale crack information in images and demonstrates enhanced robustness against occlusions. Integrated into the detection head, SOAA enables a more efficient extraction of crack features in shadowed areas. The architecture of SOAA is illustrated in Figure 7.

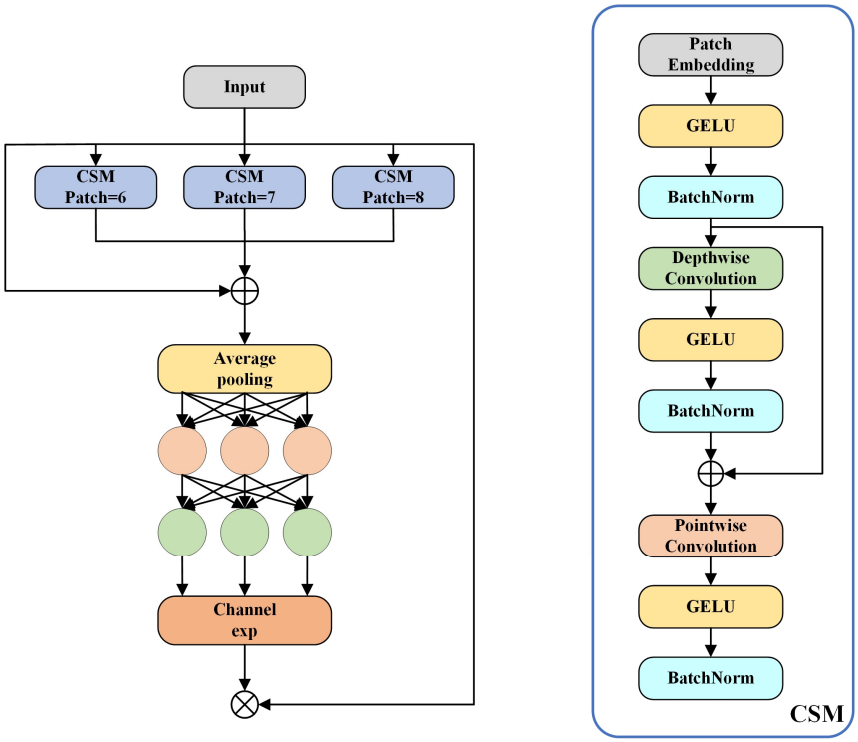


Figure 7. The structure of the SOAA module. It uses multiscale CSM blocks to extract features and enhance crack detection in shadowed regions. The right shows the internal structure of a CSM block with depthwise and pointwise convolutions.

The SOAA module primarily consists of a CSM module, average pooling operations, and fully connected layers. The CSM module employs depthwise separable convolution

with residual connections to extract crack features. Channel-wise computation reduces parameter redundancy while enhancing local crack feature capture capabilities. The subsequent pointwise convolution integrates cross-channel information, improving the learning of correlations between blurred and clear cracks. After merging features from different depthwise convolutions, an average pooling layer reduces feature dimensionality while preserving critical information. A two-layer fully connected network further consolidates multiscale relationships and enhances dynamic focus on shadow-occluded regions. As illustrated in the heatmaps of Figure 8, under extremely low-light conditions where the original YOLOv8 barely detected cracks, the integration of SOAA significantly improved the model's attention to crack regions, demonstrating the module's effectiveness in achieving high-precision crack localization in complex backgrounds.

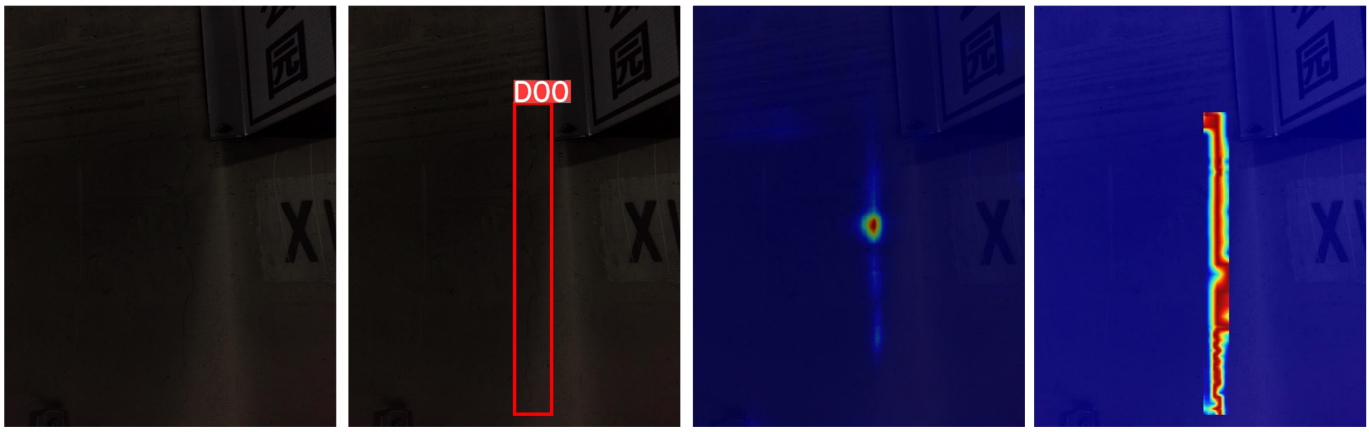


Figure 8. Thermogram of pristine YOLOV8 with addition of SOAA.

3. Experiments and Results

3.1. Experimental Data

The dataset in this paper consists of 1268 photos taken with a smartphone in various subway tunnels of a city, with a resolution of 1920 × 2560. These photos cover different subway lines and sections, showcasing the geological structures, lighting conditions, and environment within the tunnels. The tunnel crack dataset contains 983 transverse cracks (D00), 871 longitudinal cracks (D10), and 201 alligator cracks (D20). This paper categorized these images into scenes, mainly including the following: those with strong lighting, weak lighting, stain interference, obstruction interference, and shadow occlusion. An example dataset is shown in Figure 9.



Figure 9. Metro tunnel crack dataset.

3.2. Experimental Parameter Setting

The settings for the model training parameters are shown in Table 1. The server used for the model training in this study is equipped with a CPU, Intel(R) Core(TM) i7-9700, and a GPU, Nvidia GeForce RTX 4060Ti. All models were implemented using the Pytorch 2.0.1 framework.

Table 1. Parameter setting of STCYolo.

Typology	Parameters
Initial learning rate	1×10^{-5}
Batch size	8
Optimizer	Adam
Epoch	300

3.3. Model Comparison

To compare the performance of the method proposed in this paper with that of other mainstream networks, YOLOV8, DETR [23], CenterNet [24], EfficientDet [25], CrackY-OLO [26], and the model developed in this paper were trained using the dataset provided herein. Table 2 presents the recognition results of each model. The experimental results show that compared to EfficientDet, STCYOLO achieved a 12.07% increase in precision, a 12.97% increase in recall, a 13.51% increase in the F score, and an 11.61% increase in mAP. Compared to CeterNet, STCYOLO showed a 10.5% improvement in precision, a 10.15% improvement in recall, a 10.01% improvement in the F score, and a 9.72% improvement in mAP. Compared to DETR, STCYOLO exhibited a 5.04% improvement in precision, a 4.59% improvement in recall, a 5.42% improvement in the F score, and a 4.18% improvement in mAP. Compared to YOLOV8, STCYOLO achieved a 2.65% increase in precision, a 2.35% increase in recall, a 3.02% increase in the F score, and a 2.85% increase in mAP. Compared with CrackYOLO, STCYOLO improved its accuracy by 2.05%, recall rate by 2.04%, F score by 2.87%, and mAP by 1.86%.

Table 2. Indicators of model identification dataset.

Metric	EfficientDet	CenterNet	DETR	YOLOV8	CrackYOLO	STCYOLO
Precision (%)	67.32	68.89	74.35	76.74	77.34	79.39
Recall (%)	63.33	66.15	71.71	73.95	74.26	76.30
F score (%)	65.13	68.63	73.22	75.62	75.77	78.64
mAP (%)	67.22	69.11	74.65	75.98	76.97	78.83
GFLOPs (G)	46.99	70.22	208.92	81.50	79.30	78.20

In the first column of Figure 10, all models missed the minor cracks on the right side, with only the model presented in this paper accurately identifying all cracks. From the second column, it is observed that under shadow interference, DETR and CenterNet fail to identify cracks within shadows, and even cracks without shadow occlusion were not detected. Due to the addition of an antiocclusion head in this paper, our method accurately extracted cracks at shadow occlusion locations. In the third column, due to interference from brightness and background, all models except for the one presented in this paper made erroneous identifications, recognizing most of the background as alligator cracks, with only our model correctly identifying the cracks. In the fourth column, the two minor cracks above the pipeline were not identified by other networks. Due to the TCU method added in this paper, minor cracks were not ignored during the upsampling process, and our model correctly identified the cracks at that location. In the fifth column, there is a scenario with very poor lighting conditions, making it difficult even for the human eye to discern

the crack. Except for our model, which identified the crack, other models failed to extract it, whereas our model completely extracted it. Our model outperforms other mainstream models, particularly achieving higher performance in scenarios with insufficient lighting and shadows, proving the effectiveness of the algorithm presented in this paper.

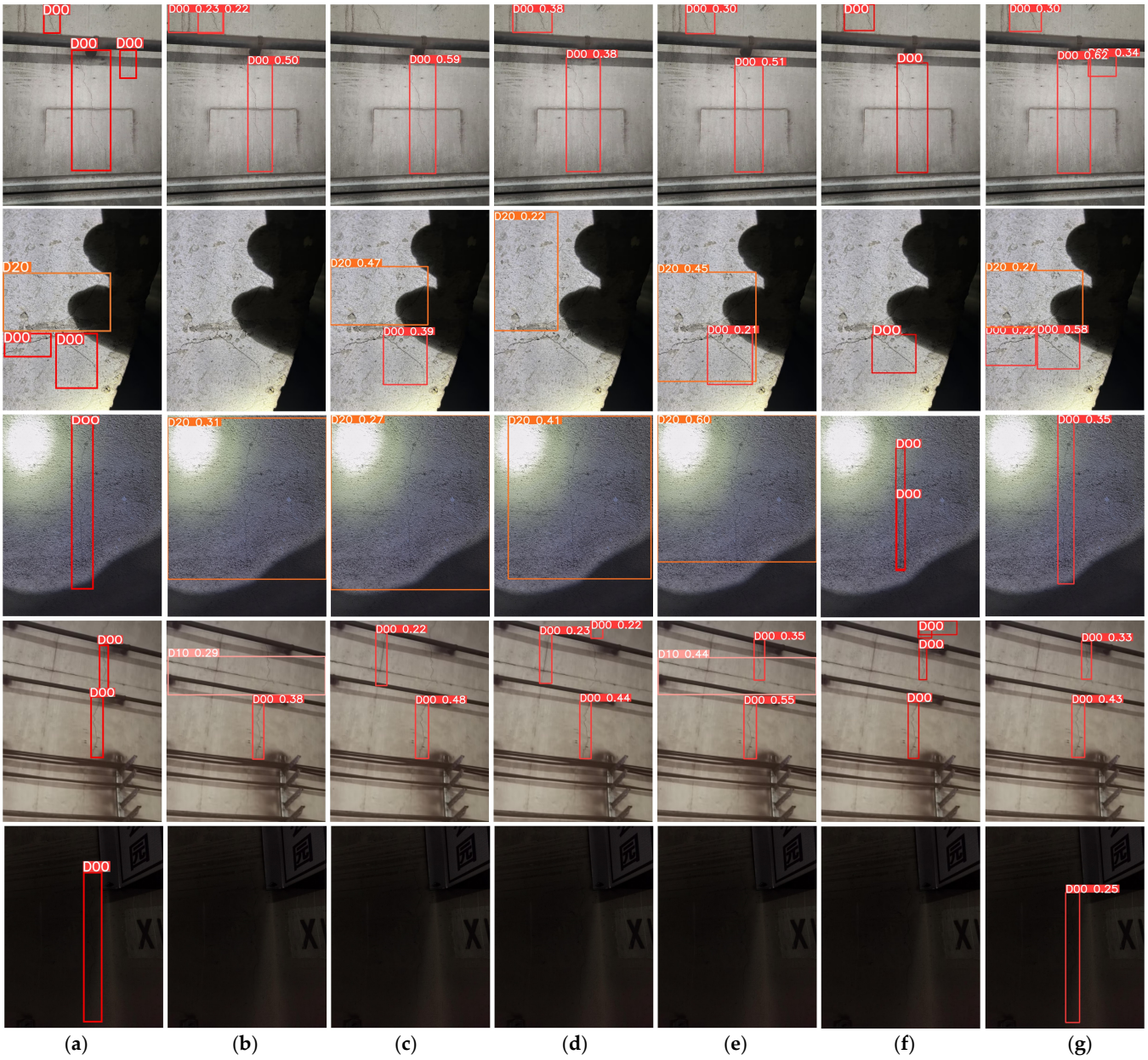


Figure 10. Results for each model recognition test set. (**a**) Ground True, (**b**) CenterNet, (**c**) EfficientDet (**d**) DETR, (**e**) YOLOV8, (**f**) CRACKYOLO, (**g**) STCYOLO.

3.4. Ablation Experiment

To verify the effectiveness of the model method presented in this paper, five sets of ablation experiments were designed: (1) YOLOV8; (2) YOLOV8 + C2f_DSConv, which represents the addition of C2f_DSConv convolution to the first layer of the YOLOV8 backbone; (3) YOLOV8 + TCU, which denotes the replacement of YOLOv8's upsampling operation with the TCU algorithm presented in this paper; (4) YOLOV8 + SOAA, which represents the replacement of YOLOV8's detection head with the SOAA introduced in this paper; and (5) STCYOLO, as proposed in this paper. Figure 11 shows the results of identifying actual subway tunnel cracks.

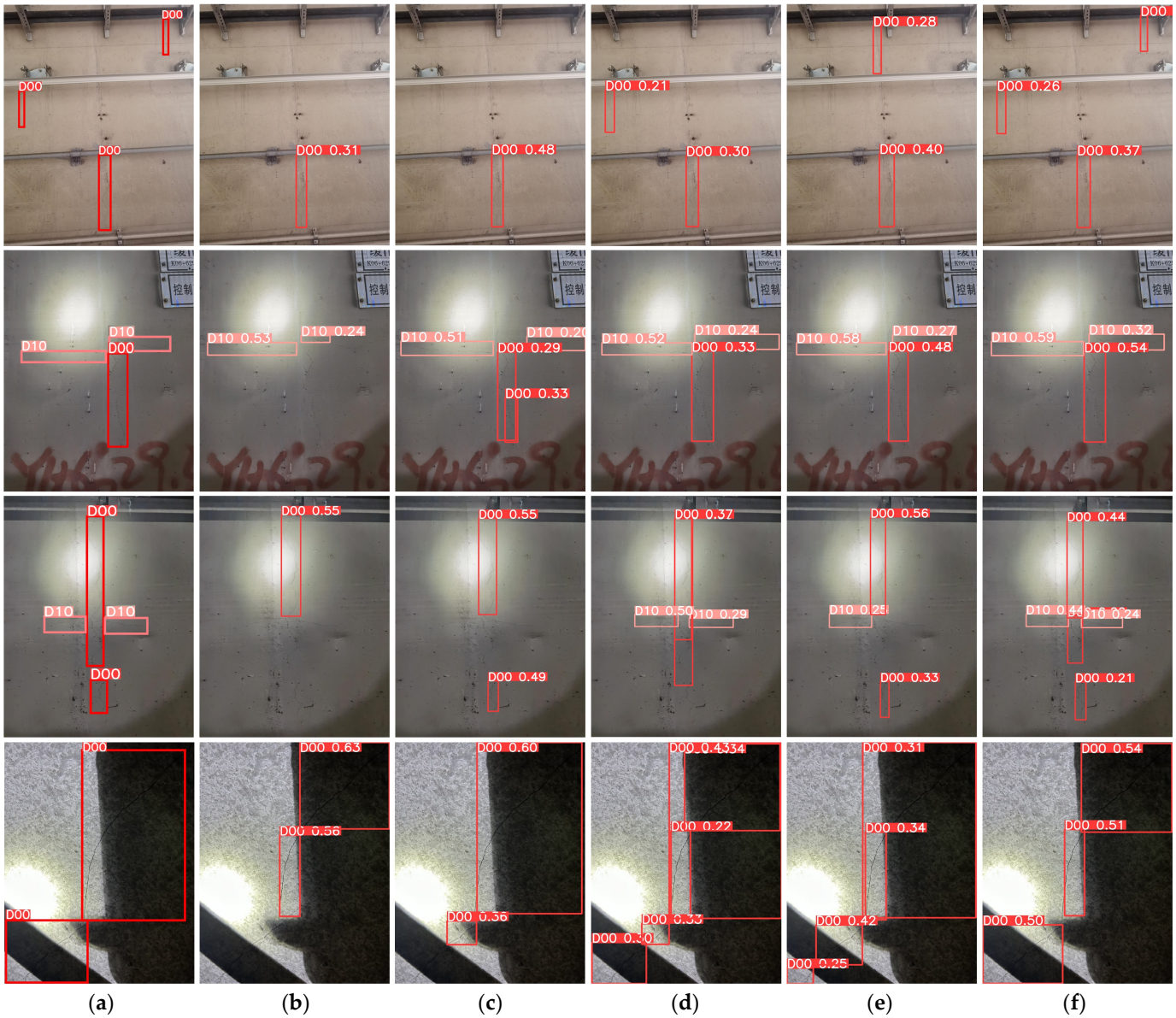


Figure 11. Adding different models to recognize test set results. (**a**) Ground True, (**b**) YOLOV8, (**c**) C2f_DSConv, (**d**) TCU, (**e**) SOAA, (**f**) STCYOLO.

In the first row, the original YOLOv8 detected only a single prominent crack with low confidence. With C2f_DSConv, detection confidence increased significantly; with TCU, additional fine cracks on the right were successfully identified; with SOAA, the performance in shadowed regions improved. In the second row, YOLOv8 missed vertical cracks entirely. Each proposed module helped extract different structural details, with SOAA being especially effective under occlusion. In the third row, cracks in low-light areas were not detected by YOLOv8. Both C2f_DSConv and TCU improved the continuity and completeness of crack detection across lighting variations, while SOAA significantly enhanced feature extraction in dim regions. In the fourth row, YOLOv8 failed to detect a shadow-obscured crack in the lower-left corner. TCU improved the extraction of subtle cracks, while SOAA overcame the shadow interference and correctly detected the missing crack.

The quantitative results in Table 3 confirm that each module independently contributes to performance enhancement. STCYOLO, which combines all three modules, delivers the most robust and comprehensive detection results under complex tunnel conditions.

Table 3. Predicted test set accuracy for each module.

Baseline	C2f_DSConv	TCU	SOAA	mAP (%)	GFLOPs (G)
✓				75.98	79.30
	✓			76.22	80.70
		✓		76.84	79.70
			✓	77.92	75.20
	✓	✓	✓	78.83	78.20

To determine the optimal deployment of the C2f_DSConv module within the YOLOv8 backbone, we conducted experiments by progressively replacing the original C2f layers at four positions. The baseline YOLOv8 achieved a mAP of 75.98% with a computational cost of 78.9 GFLOPs. When only the first C2f module was replaced, the mAP slightly increased to 76.07%, with GFLOPs rising to 80.0. Replacing the first two C2f modules yielded a more notable improvement, achieving a mAP of 76.22% and 80.3 GFLOPs. With the first three modules replaced, the mAP reached 76.39% at 81.3 GFLOPs, and the full replacement of all four C2f modules resulted in a mAP of 76.25% with 81.8 GFLOPs. Although deeper replacement led to marginal accuracy gains, it also introduced considerable computational overhead. Therefore, considering the balance between accuracy improvement and model efficiency, we adopt the configuration that replaces the first two C2f modules with C2f_DSConv. The detailed results are presented in Table 4.

Table 4. Impact of replacing different layers of C2f with C2f_DSConv on model.

Baseline	1	2	3	4	mAP (%)	GFLOPs (G)
✓					75.98	79.30
	✓				76.07	80.00
	✓	✓			76.22	80.70
	✓	✓	✓		76.24	81.70
	✓	✓	✓	✓	76.25	82.20

4. Discussion

Although the model in this paper improves the detection accuracy of tunnel cracks to a certain extent, there are still some deficiencies in the model, This section discusses the limitations of the proposed algorithm and identifies areas for future improvement.

4.1. The Complexity of the Model

While the addition of multiple modules enhances the model's capability in feature extraction and robustness, it inevitably increases structural complexity and computational cost. This may pose challenges in real-time scenarios, where excessive computation can lead to system delays and reduced responsiveness. Nevertheless, as shown in Table 4, the overall GFLOPs of the proposed STCYOLO model (78.20 G) remain significantly lower than those of DETR (208.92 G) and comparable to CenterNet (70.22 G) while achieving superior detection accuracy. This indicates that the proposed model maintains a favorable balance between detection performance and computational efficiency. In future work, further optimization techniques—such as model pruning and weight sharing—could be explored to reduce model size and improve inference speed without compromising accuracy.

4.2. Dataset

The current model demonstrates strong detection capabilities on the training dataset; however, its performance may be sensitive to the distribution and characteristics of the training data. In real-world tunnel environments, Variations in lighting, structural features,

and surface texture can result in significant deviations from the training conditions. which may affect generalization. As shown in Figure 10, although STCYOLO achieves consistent results across diverse tunnel scenes, a slight drop in performance is observed when tested on a dataset with significantly different illumination, suggesting a degree of domain dependence. This indicates that while the model architecture is robust, further improvements in data diversity and domain adaptation are necessary. In future work, techniques such as data augmentation, domain adaptation, and transfer learning could be employed to narrow the distribution gap between training and deployment environments and enhance model robustness.

4.3. The Shortcomings of the SOAA Module

The Shadow Occlusion-Aware Attention (SOAA) module was introduced to address the challenge of crack detection in shadowed or partially occluded regions. This component enhances the model's ability to focus on relevant features under complex lighting conditions. As demonstrated in Figure 10 (column 2 and 5), STCYOLO outperforms baseline models in scenes with heavy shadow interference, effectively identifying cracks that other models fail to detect. This confirms the effectiveness of SOAA in real tunnel environments. However, under extreme conditions such as severe underexposure or strong specular reflection, some crack features may still be obscured. To mitigate this, future improvements may incorporate multiscale attention mechanisms or image preprocessing techniques to further boost the robustness of crack feature extraction.

4.4. The Shortcomings of the TCU Module

The TCU module is designed to enhance the detection of fine, low-contrast cracks by preserving detail during upsampling. As shown in Figures 10 and 11, STCYOLO successfully detects small cracks around pipeline edges and image borders that are typically missed by other models, validating the effectiveness of TCU. However, the increased sensitivity to minute features may also result in a higher false positive rate, particularly in the presence of textured backgrounds or noise. Some non-crack structures may be mistakenly classified as cracks. To address this, future designs could incorporate structural constraints or spatial continuity priors, allowing the model to better distinguish between true cracks and similar non-crack features, thereby reducing false detections while maintaining fine-scale sensitivity.

5. Conclusions

Crack detection in the complex environment of subway tunnels is easily affected by lighting, shadows, and other disturbances, leading to lower detection accuracy. To address this issue, this paper proposes the STCYOLO subway tunnel crack detection network, incorporating DSC, TCU, and SOAA head modules to tackle the problems of shadow occlusion and uneven light distribution, causing cracks to be missed. This study created a subway tunnel crack dataset for experimentation. The experimental results indicate that the STCYOLO model has a significant advantage in subway tunnel crack detection tasks compared to other detection models, demonstrating good detection performance across various scenarios. Although the model proposed in this paper has achieved promising results, it still has certain shortcomings:

(1) Adding different models result in a complex model structure and increased computational complexity. Especially in scenarios with high real-time requirements, this may lead to a bottleneck.

(2) The model may have a high dependence on specific training data, and the current data distribution may not fully match the actual application scenario, which may affect the performance of the model.
(3) The SOAA module may mainly focus on detecting cracks in shadows, but cracks may still be obscured or blurred in areas of extremely low brightness or strong reflection.
(4) The TCU module may focus on improving the detection capability of small cracks, but at the same time, this may lead to an increase in the false detection rate.

In the future, the abovementioned issues should be addressed.

Author Contributions: Conceptualization, J.Z. (Jia Zhang) and H.L.; methodology, J.Z. (Jia Zhang); validation, J.Z. (Jia Zhang) and H.L.; formal analysis, J.Z. (Jia Zhang); investigation, H.L.; resources, W.S.; data curation, W.S.; writing—original draft preparation, J.Z. (Jia Zhang); writing—review and editing, J.Z. (Jinhe Zhang). and H.L.; visualization, J.Z. (Jinhe Zhang) and M.S.; supervision, M.S.; project administration, J.Z. (Jia Zhang); funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Scientific Research Project of the Liaoning Provincial Department of Education, Grant No. LJKMZ20222194 and National Natural Science Foundation of China, under Grant 42071343.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data provided in this work are available from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yang, H.; Xu, X. Intelligent Crack Extraction Based on Terrestrial Laser Scanning Measurement. *Meas. Control* **2020**, *53*, 416–426. [CrossRef]
2. Liu, D.; Deng, Y.; Yang, F.; Xu, G. Nondestructive Testing for Crack of Tunnel Lining Using GPR. *J. Cent. S. Univ. Technol.* **2005**, *12*, 120–124. [CrossRef]
3. White, J.; Hurlebaus, S.; Shokouhi, P.; Wimsatt, A. Use of Ultrasonic Tomography to Detect Structural Impairment in Tunnel Linings: Validation Study and Field Evaluation. *Transp. Res. Rec. J. Transp. Res. Board* **2014**, *2407*, 20–31. [CrossRef]
4. Gong, Q.; Zhu, L.; Wang, Y.; Yu, Z. Automatic Subway Tunnel Crack Detection System Based on Line Scan Camera. *Struct. Control Health Monit.* **2021**, *28*, e2776. [CrossRef]
5. Li, C.; Xu, P.; Niu, L.; Chen, Y.; Sheng, L.; Liu, M. Tunnel Crack Detection Using Coarse-to-fine Region Localization and Edge Detection. *WIREs Data Min. Knowl. Discov.* **2019**, *9*, e1308. [CrossRef]
6. Jiang, F.; Wang, G.; He, P.; Zheng, C.; Xiao, Z.; Wu, Y. Application of Canny Operator Threshold Adaptive Segmentation Algorithm Combined with Digital Image Processing in Tunnel Face Crevice Extraction. *J. Supercomput.* **2022**, *78*, 11601–11620. [CrossRef]
7. Ba, Y.; Zuo, J.; Jia, Z. Image Filtering Algorithms for Tunnel Lining Surface Cracks Based on Adaptive Median-Gaussian. In Proceedings of the ICTE 2019; American Society of Civil Engineers, Chengdu, China, 13 January 2020; pp. 849–853.
8. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
9. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
10. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper With Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9.
11. Xue, Y.; Li, Y. A Fast Detection Method via Region-Based Fully Convolutional Neural Networks for Shield Tunnel Lining Defects. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 638–654. [CrossRef]
12. Zhou, Z.; Zhang, J.; Gong, C.; Wu, W. Automatic Tunnel Lining Crack Detection via Deep Learning with Generative Adversarial Network-Based Data Augmentation. *Undergr. Space* **2023**, *9*, 140–154. [CrossRef]
13. Li, D.; Xie, Q.; Gong, X.; Yu, Z.; Xu, J.; Sun, Y.; Wang, J. Automatic Defect Detection of Metro Tunnel Surfaces Using a Vision-Based Inspection System. *Adv. Eng. Inform.* **2021**, *47*, 101206. [CrossRef]

14. Juan, S.; Long-Xi, H.; Hui-Ping, L. Tunnel Lining Multi-Defect Detection Based on an Improved You Only Look Once Version 7 Algorithm. *IEEE Access* **2023**, *11*, 125171–125184. [CrossRef]
15. Dai, Q.; Xie, Y.; Xu, J.; Xia, Y.; Sheng, C.; Tian, C.; Ou, W. Tunnel Crack Identification Based on Improved YOLOv5. In Proceedings of the 2022 7th International Conference on Automation, Control and Robotics Engineering (CACRE), Xi'an, China, 14–16 July 2022; pp. 302–307.
16. Ren, Y.; Huang, J.; Hong, Z.; Lu, W.; Yin, J.; Zou, L.; Shen, X. Image-Based Concrete Crack Detection in Tunnels Using Deep Fully Convolutional Networks. *Constr. Build. Mater.* **2020**, *234*, 117367. [CrossRef]
17. Li, G.; Ma, B.; He, S.; Ren, X.; Liu, Q. Automatic Tunnel Crack Detection Based on U-Net and a Convolutional Neural Network with Alternately Updated Clique. *Sensors* **2020**, *20*, 717. [CrossRef] [PubMed]
18. Liao, J.; Yue, Y.; Zhang, D.; Tu, W.; Cao, R.; Zou, Q.; Li, Q. Automatic Tunnel Crack Inspection Using an Efficient Mobile Imaging Module and a Lightweight CNN. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 15190–15203. [CrossRef]
19. Zhou, Q.; Qu, Z.; Li, Y.-X.; Ju, F.-R. Tunnel Crack Detection With Linear Seam Based on Mixed Attention and Multiscale Feature Fusion. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 5014711. [CrossRef]
20. Zhou, Z.; Zheng, Y.; Zhang, J.; Yang, H. Fast Detection Algorithm for Cracks on Tunnel Linings Based on Deep Semantic Segmentation. *Front. Struct. Civ. Eng.* **2023**, *17*, 732–744. [CrossRef]
21. Zhu, X.; Hu, H.; Lin, S.; Dai, J. Deformable ConvNets V2: More Deformable, Better Results. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 9308–9316.
22. Qi, Y.; He, Y.; Qi, X.; Zhang, Y.; Yang, G. Dynamic Snake Convolution Based on Topological Geometric Constraints for Tubular Structure Segmentation. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 6070–6079.
23. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
24. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6568–6577.
25. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 10778–10787.
26. Li, Y.; Sun, S.; Song, W.; Zhang, J.; Teng, Q. CrackYOLO: Rural Pavement Distress Detection Model with Complex Scenarios. *Electronics* **2024**, *13*, 312. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.