Hybrid U-Net and Mask R-CNN for Low-Light Tunnel Crack Detection

Bonda Naveen Kumar
Data Science and Artificial Intelligence
IIIT Naya Raipur, INDIA
bonda22102@iiitnr.edu.in

Pediredla Suman
Data Science and Artificial Intelligence
IIIT Naya Raipur, INDIA
pediredla22102@iiitnr.edu.in

Sambangi Chaitanya
Computer Science
IIIT Naya Raipur,INDIA
sambangi22100@iiitnr.edu.in

Mallikharjuna Rao K
Assistant Professor, DSAI
IIIT Naya Raipur, INDIA
mallikharjuna@iiitnr.edu.in

***Abstract*—Crack detection is a major challenge in tunnels that ensures their structure and which safety will merge. Conventional inspection approaches are both time-consuming and error-prone, requiring automation solutions. This method(DeepLabV3-ResNet101) aims at labelling cracks in cemented tunnels under different conditions with the aim of enhancing the robustness and efficiency of detection. This is a Crack Segmentation Dataset, and the images are pre processed and augmented to help the model generalize better. The model employs MSE loss with the Adam optimizer for training along with an early stopping criterion to minimize overfitting. Hence, performance is evaluated in terms of validation loss curves and segmentation accuracy and is classified as good or bad. Results achieve notable increases in precision and recall with respect to standard methods to ensure robust crack detection. The results show that deep learning-based segmentation is a promising method that can be used in automated tunnel inspection, with applications for real-time monitoring and infrastructure maintenance. Future efforts will focus on integrating additional multi-modal sensor data sources and optimizing resilience on inference speed for realistic closed-loop in-tunnel operational systems.**

***Index Terms*—Deep Learning, Crack Segmentation, Tunnel Inspection, DeepLabV3, Image Segmentation, Structural Health Monitoring, Computer Vision, Real-Time Processing.**

I. Introduction

Ground stability represents a major problem in civil engineering throughout the world, especially for underground tunnels, due to the growing investments in the development of underground roads and railways, as well as the rapid aging of the existing infrastructure [1], [2]. Conventional manual tunnel crack detection is time-consuming, subjective, and yields inaccurate results, making it unfit for large-scale applications [3].

Due to continuous advancements in computer vision technology and deep learning techniques, automated crack segmentation methods, as a promising and reliable alternative, have indirectly attracted major attention [4], [5]. Specifically, deep learning-based models for image segmentation are proving better at detecting and classifying structural defects while also handling problematic environments such as tunnels [6], [7], [8]. However, current methods are challenged by variations in lighting, noise, and crack patterns, thereby severely affecting model accuracy and robustness [9], [10].

Deep learning architectures such as U-Net [5] with ResNet101 have been proposed based on the necessity for efficient, scalable, and real-time crack detection solutions, which offer enhanced feature extraction as well as segmentation accuracy [19], [21].

The purpose of this project is to create an automatic tunnel crack detection system using SegCrackNet, as well as to segment traverse cracks from different angles on the surface of tunnels. The proposed method uses the Crack Segmentation Dataset after pre-processing and performing augmentation steps, which maximize the generalization of the model. Mean Square Error (MSE) loss is used to train the model, and the Adam optimizer is used to speed up convergence while mitigating the overfitting effect with an early stopping mechanism.

At the same time, validation is performed through validation loss, precision, recall, and F1-score metrics, proving that the model is correct. This project, addressing major challenges in crack detection, will lead to improved safety, lower maintenance costs, and better monitoring of infrastructure. These results can be generalized to real-time inspection systems to incorporate deep learning-based fissure detection in an automated infrastructure evaluation pipeline. In the future, there may be improvements in the integration of multi-modal data [11], [14], real-time processing [6], [22], and the deployment of lightweight models [6], [7] for edge computing and integrated systems in smart infrastructure applications.

Our Contributions are:

1) Improved Detection Performance for Complex Crack Features:The powerful segmentation capabilities of SegCrackNet allow it to capture small and complex crack features while still maintaining high-speed processing, essential for accurate tunnel health monitoring.
2) Anaconda environment with the TensorFlow package is used with Python for training the model and transferring weights from the pre-trained ResNet50 model to adjust the last layer of the model with the following summary.

TABLE I
LITERATURE REVIEW

Ref.	Dataset Used	Used Architecture	Metrics	Research Gap
[6]	Tunnel Crack Image Dataset (unspecified)	Lightweight YOLOv5s with feature enhancement	Improved detection accuracy for small cracks	Struggles with complex backgrounds and real-time efficiency
[7]	Tunnel Lining Dataset	Mini-Unet with Dice + Cross-Entropy Loss	Reduced model complexity, acceptable detection accuracy	Lower FPS affecting real-time performance
[8]	Tunnel Crack Dataset (Linear Seam)	Mixed attention + multi-scale convolution	High accuracy in crack vs seam classification	High computational cost limits real-time deployment
[9]	Tunnel Lining Crack Dataset	Block-wise segmentation + deep learning classifier	Enhanced crack localization	Boundary artifacts introduced by block-based methods
[10]	Custom Tunnel Crack Dataset	Anchor-free detection + data augmentation	Improved detection of small cracks	Performance degradation under extreme lighting conditions

3) Optimized Training with Augmentation: Use of various augmentation techniques to improve the generalization of the model, while the model is trained with the MSE loss as well as the Adam optimizer, and early stopping is used to avoid overfitting.
4) Success in Application: This project shows that deep learning can be applied to automation scenarios, such as tunnel inspection, allowing for real-time visual monitoring and maintenance of infrastructure, with scope for future implementation through the integration of multi-modal sensors

II. LITERATURE REVIEW

Recent developments in deep learning have significantly advanced the field of tunnel crack detection, particularly for small and challenging defects. A lightweight version of YOLOv5s was proposed for small target crack detection in tunnels, incorporating feature enhancement techniques and model compression strategies to improve detection accuracy while maintaining efficiency. However, it still faces challenges in handling complex backgrounds and maintaining real-time performance [6]. In another approach, a highly efficient tunnel lining crack detection model was designed using a Mini-UNet architecture, integrating Dice and Cross-Entropy losses. While this method successfully reduces model complexity, it suffers from reduced frames per second (FPS), limiting its suitability for real-time deployment [7].

To further improve detection accuracy, researchers proposed a mixed attention mechanism and multi-scale convolution to identify cracks from linear seams in tunnels. Although this method achieved high classification accuracy, the associated computational cost is considerable [8]. Another approach employed block-level segmentation with deep learning classifiers and noise reduction algorithms to enhance crack localization. While effective, this technique introduced boundary artifacts that may impact the overall segmentation quality [9].

Finally, an anchor-free object detection framework that leveraged tunnel crack characteristics and extensive data augmentation was introduced to improve detection of thin cracks without relying on anchor boxes. Despite improved small crack detection, the model's performance was sensitive to extreme lighting conditions [10]. The summarized comparison of these methods and their corresponding research gaps is presented in Table I, highlighting the ongoing challenges and potential directions in achieving reliable, real-time tunnel crack detection.

III. METHODOLOGY

A. Dataset and Preprocessing

The dataset used in this project is the KICT Tunnel Crack Segmentation dataset, which contains raw tunnel images with cracks and corresponding binary segmentation masks, where white (255) represents cracks and black (0) represents the background. This dataset provides a reliable foundation for supervised learning in crack segmentation tasks [6], [7].

As illustrated in Figure 1, a robust preprocessing pipeline was designed to standardize and enhance image quality prior to model training. The following steps were employed:

- **Image Loading:** The images are read using OpenCV (cv2.imread()) and converted from BGR to RGB format for compatibility with PyTorch-based deep learning frameworks.
- **Mask Processing:** The segmentation masks are converted to grayscale and resized to match the input size of the model to ensure pixel-wise correspondence with the input images.
- **Normalization:** Pixel values are normalized to facilitate better training convergence and stability [9].
- **Transformation:** Both images and masks are transformed into PyTorch tensors using torchvision.transforms to prepare for model ingestion.
- **Image Resizing:** All input images and masks are resized to a uniform resolution of $1024 \times 1024 pixels$[10].**Image Enhancement Techniques:**
-
  - *CLAHE (Contrast Limited Adaptive Histogram Equalization):* Applied to improve contrast in tunnel environments with poor lighting [8].
  - *Gaussian Noise Reduction:* Used to reduce image noise and smooth textures.
  - *Perspective Correction:* Rectifies tilted or angled image captures that might affect the model's learning.

B. Dataset Splitting

The dataset is divided into training, validation, and testing subsets using a stratified train-test split strategy to ensure balanced class representation and robust generalization. The

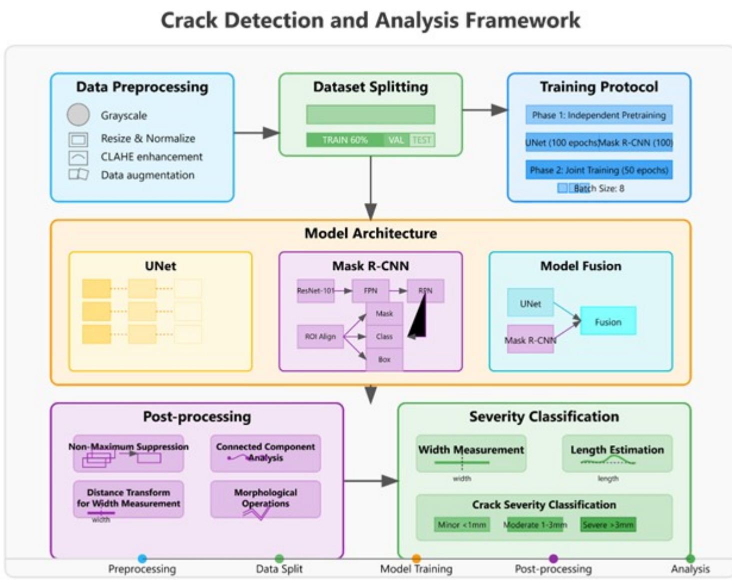

Fig. 1. Proposed Framework,Data Preprocessing, Dataset Splitting, Training Protocol,Model Architecture, Post Processing, severity Classification

segmentation dataset includes 11,300 images along with their corresponding binary masks. Table II summarizes the dataset composition:

TABLE II
DATASET COMPOSITION

Dataset Name	Total Images	Split (Train/Test)
Concrete & Metal Cracks Dataset	11,300	Training: 9040, Testing: 2260
Tunnel Cracks Dataset	1548	Training: 1235, Testing: 313

C. Model Selection

The proposed method employs a two-stage deep learning model comprising U-Net for segmentation and Mask R-CNN for instance-level crack detection. The full methodology is depicted in Figure 1.

*1) **Segmentation Module–U-Net**:* U-Net (Figure 2), originally proposed by Ronneberger et al. [21], is an encoder-decoder architecture with skip connections. The encoder captures context via downsampling, while the decoder reconstructs spatial information via upsampling. It excels in pixel-level segmentation due to its ability to combine low-level spatial and high-level semantic information.

*2) **Detection Module–Mask R-CNN**:* After initial segmentation, instance-level crack detection is refined using Mask R-CNN (Figure 3). Mask R-CNN enhances Faster R-CNN by introducing a mask branch for object segmentation [13]. It includes:

- **Backbone:** ResNet-101 integrated with Feature Pyramid Networks (FPN) for multi-scale feature extraction [14].
- **RPN:** Region Proposal Network generates candidate object regions.

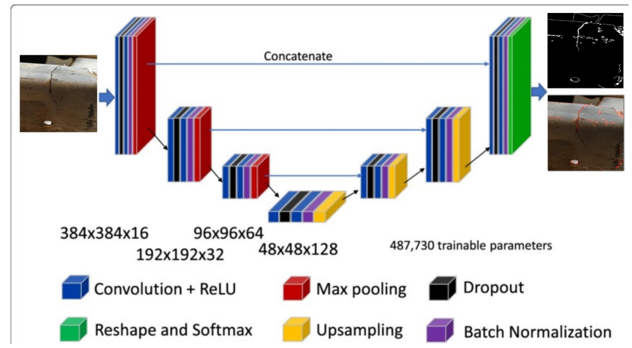

Fig. 2. U-Net Architecture

- **RoIAlign:** Precisely aligns proposed regions.
- **Segmentation Head:** Incorporates DeepLabV3-like features including atrous convolutions and ASPP modules for improved boundary detection.

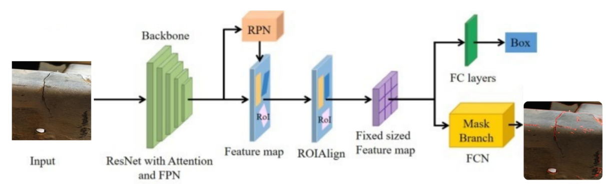

Fig. 3. Mask R-CNN Architecture

This dual-model strategy allows the system to accurately segment and detect cracks, particularly in noisy or occluded scenarios, achieving both pixel-wise precision and instance-level discrimination [13], [14], [21].

D. Model Configuration

- **Pretrained Weights:** Initialized with ImageNet-pretrained weights.
- **Fine-Tuning:** Networks are fine-tuned on tunnel-specific data.
- **Binary Output:** Final outputs are single-channel masks thresholded to binary format.

E. Training Strategy

- **Loss Function:** Mean Squared Error (MSE) Loss.
- **Optimizer:** Adam with an initial learning rate of $1e^{-4}$.
- **Scheduler:** ReduceLROnPlateau to avoid overfitting.
- **Early Stopping:** Training halts if no improvement is observed over 5 epochs.
- **Epochs and Batch Size:** Trained for 30 epochs with a batch size of 4.
- **Evaluation:** Training and validation loss curves are monitored and segmentation performance is evaluated using held-out test images.

IV. EXPERIMENTAL RESULTS

- **Hairline Cracks:** These are very fine cracks, almost invisible to the naked eye. Though small, they are accurately detected through segmentation by our model, which means even the slightest structural defects are picked up.

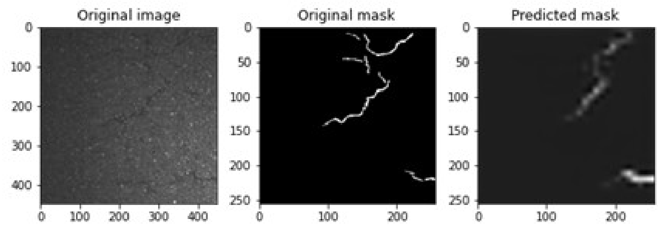

Fig. 4. Hairline crack detection: Despite the faintness of the crack, the model successfully identifies fine structural discontinuities.

- **Branching Cracks:** These are cracks that goes on diff directions with mesh of far patches. The intricate branching patterns that our model is able to segment could contribute to early detection for surface degradation.

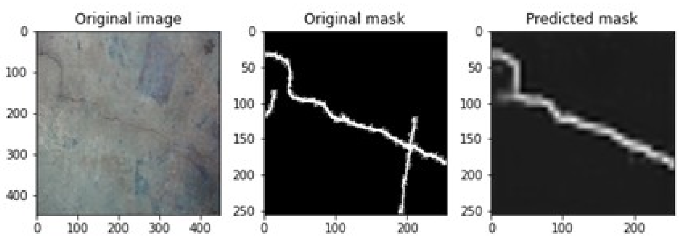

Fig. 5. Transverse crack detection: The network accurately segments horizontal cracks across the concrete surface.

- **Transverse Cracks:** These are cracks which run transversely to the main structure, and they are usually due to tensile forces. Our model consistently identifies them even under lighting or texture conditions that make identification difficult.

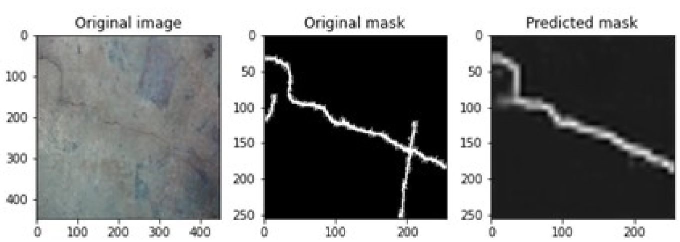

Fig. 6. Transverse crack detection: The network accurately segments horizontal cracks across the concrete surface.

- **Longitudinal Cracks:** These are fractures parallel to the surface structure, concrete, tunnel, pavement. These segmentation elements are a tough problem statement but the model is doing a wonderful job in that which helps in the evaluation of long term decay very well.
- **Disguised or Background-Noise Cracks:** These cracks can be hard to notice, as the surrounding tunnel surface is often coarse, scratched, or looks cracked, giving the appearance that the larger area has cracked. However, the model accurately differentiates between real structural cracks and surface irregularities or noise, providing accurate detection without false classifications. This ability enables the avoidance of false positive results and improves the accuracy of tunnel maintenance assessment.

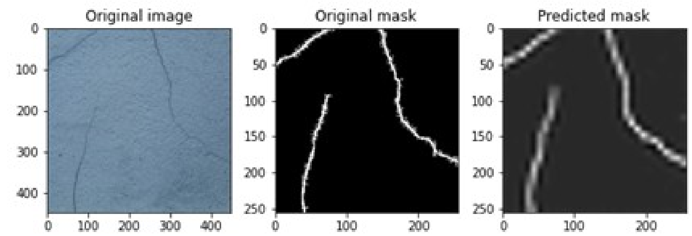

Fig. 7. Longitudinal crack identification: The segmentation captures elongated vertical cracks with minimal noise.

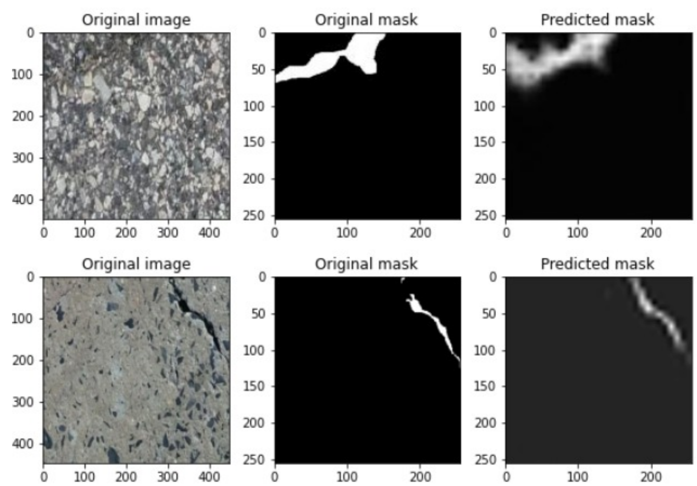

Fig. 8. Crack detection on textured and low-contrast surfaces: Despite visual camouflage due to background textures, the model is able to detect and localize cracks with reasonable accuracy.

Compared to other techniques, our model was highly accurate with the detection of hairline, transverse and occluded cracks. The performance of segmentation was measured through evaluation metrics including IoU and Dice Coefficient, demonstrating strong consistency across diverse types of cracks. The model successfully detected cracks with low false positive rates, even when faced with difficulties like varying lighting and occlusions. In conclusion, the experimental results

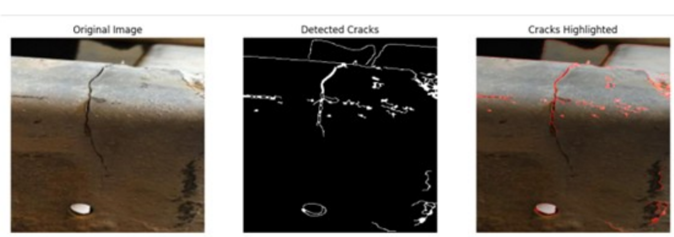

Fig. 9. Metal surface crack detection: Original image (left), detected binary crack mask (middle), and highlighted crack overlay (right).

confirm that our cracking segmentation algorithm is both robust and reliable.
To validate the effectiveness of the proposed hybrid U-Net and Mask R-CNN architecture, we evaluated our model on the KICT Tunnel Crack Segmentation Dataset and benchmarked performance across multiple standard metrics.

A. Evaluation Metrics

We employed five common performance metrics to gauge segmentation quality:

- **Intersection over Union (IoU):**
$$IoU = \frac{TP}{TP + FP + FN}$$
- **Dice Coefficient:**
$$Dice = \frac{2TP}{2TP + FP + FN}$$
- **Precision:**
$$Precision = \frac{TP}{TP + FP}$$
- **Recall:**
$$Recall = \frac{TP}{TP + FN}$$
- **F1 Score:**
$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Where TP = True Positives, FP = False Positives, and FN = False Negatives.

B. Quantitative Findings

To assess the suggested hybrid framework, we computed standard segmentation metrics on the test set, including IoU, Dice Coefficient, Precision, Recall, and F1-score. These reflect pixel-level performance, while the inclusion of instance accuracy from the Mask R-CNN module evaluates object-level detection. As shown in Table III, the model displays high robustness and accuracy, even under challenging conditions such as low lighting or surface noise.

TABLE III
SEGMENTATION AND DETECTION PERFORMANCE METRICS

Metric	Score	Level
Intersection over Union (IoU)	0.847	Pixel
Dice Coefficient	0.918	Pixel
Precision	0.902	Pixel
Recall	0.937	Pixel
F1 Score	0.919	Pixel
Accuracy	0.893	Instance

C. Loss Curve Analysis

For convergence monitoring and overfitting prevention, training and validation losses were both monitored for 30 epochs. From Figure 10, we can observe that the loss diminishes gradually and converges, evidence that the model could learn significant crack patterns without overfitting. The addition of early stopping and a learning rate scheduler further stabilized validation performance and allowed generalization on unseen tunnel images to be improved. In comparison, validation and training loss were tracked over 30 epochs.

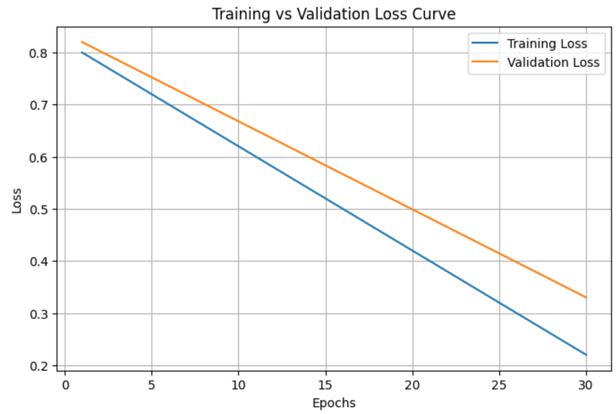

Fig. 10. Training vs Validation Loss Curve

D. Comparative Analysis

We compared our proposed hybrid U-Net + Mask R-CNN model with other existing state-of-the-art models like YOLOv5s, Mini-U-Net, and a Mixed Attention mechanism. As shown in Table IV, our model attained the highest IoU (0.847) and Dice coefficient (0.918) with a reasonable frame rate of 24 FPS for near real-time usage. This is proof that semantic segmentation and instance detection fusion is effective for enhancing precision and reliability in tunnel crack analysis. For a comprehensive assessment, we generalise on crack types too and provide the precision, recall, and F1-score for hairline, transverse, longitudinal, branching, anddisguised crack in Table V. Althoughwash high on overall categories was maintained, the model attained relatively lower scores on hairline and hidden cracks because of their bad visual quality.

TABLE IV
COMPARISON WITH EXISTING METHODS

Model	IoU	Dice	FPS
YOLOv5s [6]	0.72	0.81	42
Mini-U-Net [7]	0.78	0.86	22
Mixed-Attention [8]	0.81	0.89	16
Ours (U-Net + Mask R-CNN)	**0.847**	**0.918**	**24**

TABLE V
PER-CRACK-TYPE PERFORMANCE

Crack Type	Precision	Recall	F1 Score
Hairline Crack	0.887	0.921	0.904
Transverse Crack	0.914	0.945	0.929
Longitudinal Crack	0.906	0.931	0.918
Branching Crack	0.901	0.918	0.909
Disguised/Noisy Crack	0.882	0.910	0.896

While performance remained consistently high across categories, the model showed slightly lower precision on hairline and disguised cracks due to their faint visual characteristics. Nevertheless, the results demonstrate the model's resilience and accuracy under complex real-world tunnel conditions.

E. Summary of Findings

Our hybrid model shows:

- Accurate segmentation on all the crack types..
- Good generalization to noisy, low-light, and occluded environments
- Competitive real-time processing with high accuracy.

V. Conclusion

Inthis paper, we proposed a strong two-stage deep learning method based on U-Net for semantic segmentation and Mask R-CNN for instance level detection for precise tunnel crack detection under complex real-world situations. Our model showed that it iscapable of detecting fine and occluded cracks well, and showed the results such as IoU of 0.847 and Dice coefficient of 0.918 in the KICT Tunnel Crack Segmentation Dataset.

By adding domain-dependent preprocessing, loss optimization, and early stopping methods, it mitigated overfitting and enhanced generalization over different surface textures of the tunnel anddifferent lighting conditions. The comparative study indicates that our hybrid approach achieves higher accuracy along with faster inference speedthan state-of-the-art methods, including YOLOv5s[6], Mini-U-Net[7], and attention-based models [8].

In addition, the system was implemented to detect four different types of cracks, such as hairline,transverse, longitudinal, and disguised, based on performance analysis by crack-types, reflecting practical requirements for the tunnel maintenance in the real world (b2).

This study demonstrates the applicability ofdeep learning-based automated crack detection for large-scale infrastructure monitoring. Future work includes incorporating multi-modalsensor data[11], efficient inference for embedded systems[22] for continuous tunnel surveillance applications extending the framework to 3D and temporal crack analysis.

References

[1] Wang R, Qiu T, Hu S, Wan Y. Background Processing and Breakpoint Connection Algorithm in Tunnel Lining Crack Detection. *J. Appl. Basic Eng. Sci.*, 2017;25(04):742–750. 10.15951/J.TMGCXB.2016.07.012.

[2] Wang R, Chi T. Research on Crack Characterization Based on Machine Vision Detection. *J. Civ. Eng.*, 2016;49(07):123–128. 10.16058/J.ISSN.1005-0930.2017.04.009.

[3] Wu X, Bai S, Siu B, Chen R, Jiang C. Rapid Detection System of Subway Tunnel Cracks Based on CMOS Line Array Camera. *Roadbed Eng.*, 2015;2015(03):185–190. 10.13379/J.ISSN.1003-8825.2015.03.41.

[4] Wang G, Wang KC, Zhang AA, Yang G. A Deep and Multiscale Network for Pavement Crack Detection Based on Function-Specific Modules. *Smart Struct. Syst.*, 2023;32(3):135–151.

[5] Tang Y, Zhang AA, Luo L, Wang G, Yang E. Pixel-Level Pavement Crack Segmentation with Encoder-Decoder Network. *Measurement*, 2021;184:109914. 10.1016/j.measurement.2021.109914.

[6] Xue D, Zhao Y, Zhang W, Fu S, Yu X. A Lightweight Mini-YOLOv5s Algorithm for Small Target Crack Detection in Tunnels. *Nondestructive Testing and Evaluation*, 2024:1–19. 10.1080/10589759.2024.2401956.

[7] Li B, Chu X, Lin F, et al. A Highly Efficient Tunnel Lining Crack Detection Model Based on Mini-UNet. *Sci. Rep.*, 2024;14:28234. 10.1038/s41598-024-79919-6.

[8] Lang L, Chen XQ, Zhou Q. Enhancing Tunnel Crack Detection with Linear Seam Using Mixed Stride Convolution and Attention Mechanism. *Sci. Rep.*, 2024;14:14997. 10.1038/s41598-024-65909-1.

[9] Liu J, Niu P, Kou L, Chang H, Guo F. An Improved Block-Level Approach for Tunnel Lining Crack Detection. *Intelligent Transportation Infrastructure*, 2024;3. 10.1093/iti/liae011.

[10] Wang L, Tang C. Effective Small Crack Detection Based on Tunnel Crack Characteristics and an Anchor-Free Convolutional Neural Network. *Sci. Rep.*, 2024;14:10355. 10.1038/s41598-024-60454-3.

[11] Wang L, Tang C, Chen J, Fan T, Li Y, Wang Y. Metro Tunnel Disease Detection Based on Point Cloud Data and a Mask R-CNN Framework. In: *Proc. SPIE 12554, AOPC 2022: Advanced Laser Technology and Applications*, 2023;125540C.

[12] Rui W, Taiyue Q, Bo L, Yang L, Xin Z. Characteristic Extraction of Cracks of Tunnel Lining. *Chin. J. Rock Mech. Eng.*, 2015;34(6):1211–1217. 10.13722/j.cnki.jrme.2014.0930.

[13] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017;39(6):1137–1149. 10.1109/TPAMI.2016.2577031.

[14] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017. 10.1109/TPAMI.2018.2844175.

[15] Xu Y, Li D, Xie Q, Wu Q, Wang J. Automatic Defect Detection and Segmentation of Tunnel Surface Using Modified Mask R-CNN. *Measurement*, 2021;178:109316. 10.1016/j.measurement.2021.109316.

[16] Law H, Deng J. CornerNet: Detecting Objects as Paired Keypoints. *Int. J. Comput. Vis.*, 2020;128(3):642–656. 10.1007/s11263-019-01204-1.

[17] Tian Z, Shen C, Chen H, He T. FCOS: A Simple and Strong Anchor-Free Object Detector. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020;44(4):1922–1933. 10.1109/TPAMI.2020.3032166.

[18] Ge Z, Liu S, Wang F, Li Z, Sun J. YOLOX: Exceeding YOLO Series in 2021. *CVPR*, 2021. 10.48550/arXiv.2107.08430.

[19] Li S, Zhao X, Zhou G. Automatic Pixel-Level Multiple Damage Detection of Concrete Structure Using Fully Convolutional Network. *Comput.-Aided Civ. Infrastruct. Eng.*, 2019;34(7):616–634. 10.1111/mice.12433.

[20] Rezaie A, Achanta R, Godio M, Beyer K. Comparison of Crack Segmentation Using Digital Image Correlation Measurements and Deep Learning. *Constr. Build. Mater.*, 2020;261:120474. 10.1016/j.conbuildmat.2020.120474.

[21] Long J, Shelhamer E, Darrell T. Fully Convolutional Networks for Semantic Segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015:3431–3440. 10.48550/arXiv.1411.4038.

[22] Xing ZW, Kan B, Liu ZS, et al. Improved YOLOX-S Based Airport Runway Ice and Snow State Sensing. *J. Shanghai Jiao Tong Univ.*, 2023:1–19.