IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 July 2024, accepted 19 September 2024, date of publication 30 September 2024, date of current version 18 November 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3469951

RESEARCH ARTICLE

Application of Mask R-CNN and YOLOv8 Algorithms for Concrete Crack Detection

YONGJIN CHOI[1], BYONGKYU BAE[2], TAEK HEE HAN[3], AND JAEHUN AHN[2]
[1]Department of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
[2]Department of Civil and Environmental Engineering, Pusan National University, Geumjeong-gu, Busan 46241, Republic of Korea
[3]Ocean Space Development and Energy Research Department, Korea Institute of Ocean Science and Technology, Busan 49111, Republic of Korea

Corresponding author: Jaehun Ahn (jahn@pusan.ac.kr)

This work was supported in part by Korea Institute of Marine Science and Technology Promotion (KIMST) funded by the Ministry of Oceans and Fisheries under Grant 20220364, and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2022R1I1A3069043.

ABSTRACT The efficient and accurate detection of cracks in concrete structures is critical for maintaining structural integrity and safety. This study compares two state-of-the-art convolutional neural network (CNN) models, Mask R-CNN and YOLOv8, for automated concrete crack detection, each model representing two mainstream approaches for object detection and instance segmentation: single-stage and two-stage approach. We evaluate both models on 1,203 concrete images with 7:2:1 training, testing, and validation split, and assess their accuracy and processing speed. Mask R-CNN achieves a mean Intersection over Union (IoU) of 96.5% with a minimum IoU of 77% and higher consistency, compared to YOLOv8's 90.6%, which often shows complete failure with IoU of 0%. In terms of computation speed, YOLOv8 shows 0.3225 s of average processing time per image, slightly outperforming the speed of Mask R-CNN, 0.4867 s. Despite YOLOv8's faster processing speed, considering the characteristics of concrete crack detection tasks where accuracy should be prioritized over speed, Mask R-CNN seems a more proper model for reliable crack detection. We also show the accuracy of Mask R-CNN for crack detection tasks can be further enhanced by employing the ResNeXt backbone.

INDEX TERMS Crack detection, mask R-CNN, YOLO, object detection, instance segmentation.

I. INTRODUCTION

The detection of cracks in concrete structures is a critical aspect of structural health monitoring, directly influencing the safety and longevity of infrastructure. Accurate identification and assessment of cracks enable timely maintenance and repairs, preventing catastrophic failures and ensuring structural integrity. Traditionally, crack detection has relied on manual inspection and labeling, which is not only time-consuming and labor-intensive but also prone to human error and inconsistency.

In recent years, advancements in artificial intelligence (AI) have led to the development of automated crack detection methods using convolutional neural networks (CNNs) [1], [2], [3]. These AI-based approaches offer great improvements in efficiency for crack detection tasks without human involvement with high accuracy. Object detection and instance segmentation [4], [5], two key techniques in computer vision, have been increasingly applied to crack detection in concrete structures.

Object detection involves identifying and localizing objects within an image, whereas instance segmentation extends this by delineating the exact shape and boundaries of each detected object. Broadly, there are two approaches for conducting these tasks: single-stage approaches and two-stage approaches [6]. Single-stage approaches, such as those based on the YOLO (You Only Look Once) models [6], process the image in a single forward pass of the network. They are typically designed for fast processing and are suitable for real-time applications. In contrast, two-stage methods, like those based on the R-CNN (Regional Convolutional Neural Network) models [7], [8], first output candidate regions of interest (ROI), and the subsequent networks perform object detection and segmentation. They

The associate editor coordinating the review of this manuscript and approving it for publication was Mauro Fadda.

generally offer higher accuracy by refining object proposals in a separate step but may require more computation than the single-stage approach [6].

The previous studies [3], [9], [10], [11], [12], [13], [14], [15] have demonstrated the successful application of conventional CNN-based image processing models, such as fully connected convolutional networks (FCNs) [3], [11], U-Net [9], [10], [12], VGG [13], [14], [15], ResNet [13], [14], for crack detection tasks. More recently, studies [5], [16], [17], [18], [19], [20], [21] have used advanced CNN-based architectures, such as faster R-CNN [17], [22], [23], Mask R-CNN [17], [18], [23], and YOLO [16], [17], [20], [21], [23], for improved detection performance.

While previous studies show successful applications of YOLO and Mask R-CNN for concrete crack detection and segmentation, comparisons between the latest iterations of single-stage and two-stage approaches are lacking. Although some studies [17], [23] compare Mask R-CNN's performance to YOLO, they focused on the previous iterations of the YOLO published before 2022, which were limited in performance and features like instance segmentation. In addition, existing studies that use Mask R-CNN [17], [18], [23] are based on ResNet [24] as their backbone, which extracts features from images, although recent research [7] suggests that RexNeXt [25]-based backbone performs better for general-purpose instance segmentation tasks.

This paper aims to compare the performance of two state-of-the-art CNN models for concrete crack detection tasks: YOLOv8 and Mask R-CNN. Additionally, we use ResNeXt for the backbone of Mask R-CNN instead of ResNet, which is typically used in previous studies for concrete crack detection, to achieve performance improvements. We investigate these models' effectiveness in terms of detection accuracy and processing speed, providing insight into the validity of the advanced CNN models for crack detection. We also present the improved performance of Mask R-CNN with ResNeXt backbone.

II. METHOD

We use the Mask R-CNN [7] and YOLOv8 [26] model for concrete crack detection and compare their performance. The models have a fundamental architectural difference. Mask R-CNN employs a two-stage detection approach: first, it generates region proposals that include candidate object regions, and then it refines these proposals to produce object classifications, bounding boxes, and segmentation masks.

On the other hand, YOLOv8 utilizes a single-stage approach, where object detection, classification, and segmentation mask generation are performed in a single forward pass of the network. This streamlined approach enables faster processing speeds, making YOLOv8 suitable for real-time applications, albeit potentially with a trade-off in accuracy for complex scenes.

The choice between these architectures depends on the specific requirements of the application, balancing factors such as processing speed, accuracy, and the complexity of the scenes being analyzed. We explain the details of the two models in the following sections.

A. MASK R-CNN

Mask R-CNN (Regional Convolutional Neural Network) is designed for instance segmentation tasks based on CNN. It extends the capabilities of Faster R-CNN [27] by improving Region of Interest (ROI) processing through the introduction of the region of interest (ROI) alignment, and by adding a branch for predicting segmentation masks on each ROI. The Mask R-CNN framework operates in two distinct stages (Figure 1): region proposal and instance segmentation with object detection. The following sections explain each stage.

1) REGION PROPOSAL

The region proposal stage (Figure 1a and b) identifies candidate ROIs in the image that are likely to contain objects, thereby letting the subsequent processing focus on these areas. The region proposal network first processes the input image through a convolutional backbone network (Figure 1a) to extract features from the image. Typically, a variant of ResNet integrated with a Feature Pyramid Network (FPN) [28] serves as the backbone architecture [7], [27]. The FPN enhances the backbone by creating a multi-scale feature pyramid, which allows the network to detect objects at various scales effectively. The FPN achieves this by constructing feature maps at multiple scales from the ResNet. It uses a top-down pathway through lateral connections with the feature map from the ResNet, in order to merge high-level semantic features with low-level, high-resolution features. This multi-scale representation enables the network to detect both large and small objects efficiently.

In this study, we use ResNeXt [25] as our backbone architecture instead of the more commonly used ResNet in concrete crack detection studies [17], [18], [23] to obtain a potential performance improvement. ResNeXt enhances ResNet by using multiple simpler pathways located parallelly within each layer, allowing it to learn more diverse features without making the overall model much larger or more complex. He et al. [7] demonstrated that RexNeXt delivers better inference results in various object detection tasks.

Following feature extraction, a Region Proposal Network (RPN) (Figure 1b) operates on the feature map outputted from the backbone. It employs a small convolutional network that slides over the feature map and conducts evaluation. At each location, the RPN uses a set of predefined anchor boxes of different scales and aspect ratios to generate these proposals that cover a diverse range of potential object shapes and sizes. For each anchor box, the RPN returns the predicted objectness score which indicates the likelihood of the object being present, and the adjusted anchor box coordinates.

2) INSTANCE SEGMENTATION WITH BOUNDING BOX

The second stage (Figure 1c and d) refines the proposals generated in the first stage. It outputs the bounding boxes for

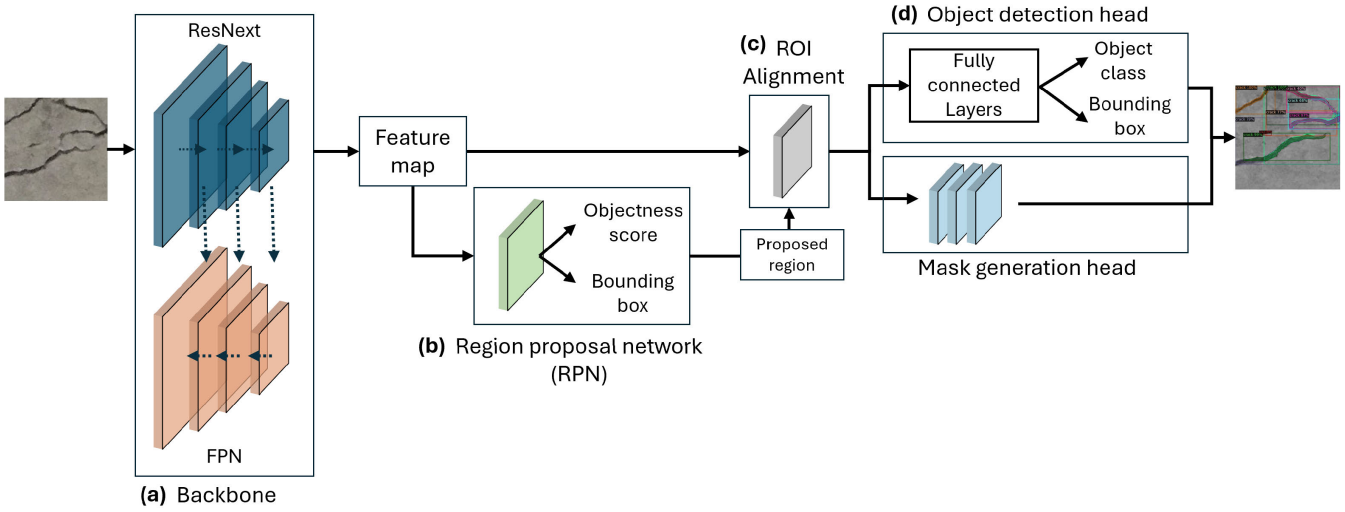

FIGURE 1. Structure of Mask R-CNN algorithm for concrete crack detection.

object detection and instance segmentation masks, which in our case represent the area of the crack.

It begins with the ROI Align layer (Figure 1c), which extracts fixed-size feature maps from each proposed region. ROI Align improves upon its predecessor, ROI Pooling, which is used in Faster R-CNN. By employing bilinear interpolation, ROI Align preserves spatial information with much greater accuracy. This prevents the quantization issue in ROI pooling, where the continuous coordinates of the proposed regions are rounded to discrete bin locations.

The aligned feature maps are then processed by two network heads (Figure 1d): the object detection head and the mask generation head. The object detection head conducts the bounding box regression and object classification using fully connected layers. The mask generation head generates a class-specific binary mask for each bounding box. This mask delineates the shape of the object, such as the crack region in our case, within the bounding box.

B. YOLOv8

YOLOv8 [26] is one of the recent versions of the YOLO (You Only Look Once) series of object detection models, which are designed for efficient object detection and instance segmentation tasks. YOLO operates as a single-stage detector, processing the entire image in a single forward pass to predict bounding boxes, class probabilities, and segmentation masks simultaneously. This is the key difference from the two-stage approach like Mask R-CNN, where the input image first goes through RPN before conducting object detection and segmentation tasks, often resulting in additional computational costs.

The components of YOLO include the backbone, neck, and head. The backbone, based on a cross-stage partial networks (CSPDarkNet) structure [29], [30], extracts features from the input image through a series of convolutional layers with Path Aggregation Network (PANet) architecture [31]. PANet is an improved architecture from FPN to enhance information flow between multi-level feature maps in the backbone. This PANet part is explicitly denoted as a "neck" in YOLO. The head of YOLO generates the final predictions. It uses convolutional layers to process the feature maps from the neck, outputting bounding boxes, class probabilities, and segmentation masks for each detected object.

A key improvement in YOLOv8 is its anchor-free detection approach [32]. Unlike earlier YOLO versions that relied on predefined anchor boxes, YOLOv8 directly predicts the center points of objects along with their dimensions. This approach simplifies the detection process and potentially improves performance on small objects.

The single-stage architecture, which does not employ the RPN, allows YOLO to achieve a faster inference with simple architecture compared to the two-stage process, making it suitable for applications requiring real-time or near-real-time performance in object detection and instance segmentation tasks, with potential accuracy trade-off.

C. TRAINING AND EVALUATION

1) TRAINING

This study utilizes a dataset of 1,203 concrete images with cracks [33]. The dataset was divided into training (70%), testing (20%), and validation (10%) sets, resulting in 842, 241, and 120 images respectively. This split ensures adequate data for model learning while reserving samples for performance assessment and fine-tuning. An early stopping function is implemented to prevent overfitting, halting the training process when validation loss increases while training loss continues to decrease.

The loss function for Mask R-CNN ($L_{\text{Mask R-CNN}}$) (eq. 1) incorporates separate loss functions for the RPN (L_{RPN}) and the R-CNN stage ($L_{\text{R-CNN}} + L_{\text{mask}}$). These loss functions

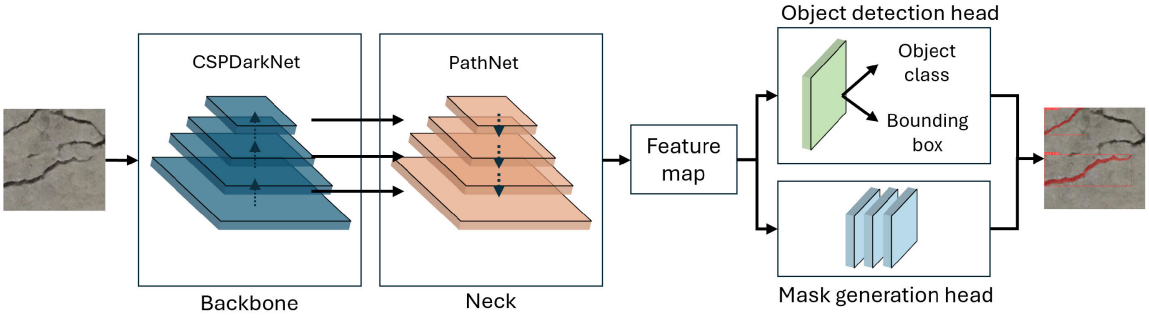

FIGURE 2. Structure of YOLO for crack detection.

are designed to handle objectness ($L_{\text{objectness}}$), classification (L_{cls}), bounding box regression (L_{bbox}), and mask predictions (L_{mask}). The entire loss function for Mask R-CNN is formulated as eq. (1) and eq. (2).

$$L_{\text{Mask R-CNN}} = L_{\text{RPN}} + L_{\text{R-CNN}} + L_{\text{mask}} \tag{1}$$

$$L_{\text{RPN}} = L_{\text{objectness}} + L_{\text{bbox}}^{\text{RPN}}$$

$$L_{\text{R-CNN}} = L_{\text{cls}} + L_{\text{bbox}}^{\text{R-CNN}} \tag{2}$$

$L_{\text{objectness}}$ measures the likelihood of the object's existence in a proposed region. L_{cls} evaluates the accuracy of the predicted class probabilities against the true class labels. L_{bbox} measures the precision of the predicted object region represented by the bounding box. L_{mask} evaluates the pixel-level accuracy of the predicted masks against the ground truth masks, which delineates the object shape. Similarly, the loss function of YOLOv8 includes classification, bounding box regression, and mask prediction losses, as given by eq. (3), but it does not have L_{RPN}.

$$L_{\text{YOLO}} = L_{\text{objectness}} + L_{\text{cls}} + L_{\text{bbox}} + L_{\text{mask}} \tag{3}$$

2) EVALUATION

The ground truth crack is defined as the area encapsulated by the crack label (Figure 3a). In Mask R-CNN, the object detection head returns the bounding box for each individual object (see the colored boxes in Figure 3b), while the mask generation head produces instance segmentations for the corresponding bounding boxes (see the colored areas in Figure 3b). Depending on the bounding box location, the mask areas may overlap. We post-process the overlapping mask areas by integrating them into a single union area by setting their RGB to the same values to define the final crack prediction (Figure 3c).

In YOLOv8, rather than the model returning the individual object and its mask area, it has a single class for the crack object, and the masked areas do not overlap. Therefore, the prediction result does not require the post-processing needed for Mask R-CNN. The predicted mask area is directly used as the final prediction.

For the quantitative performance evaluation of the models, we use the intersection over union (IoU) as the metric. The IoU (Figure 4) represents the ratio of the interaction between ground truth and predicted area over the union of the ground truth and predicted area.

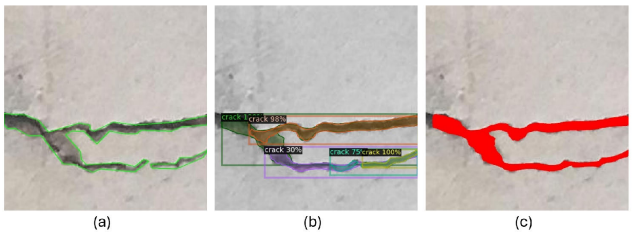

FIGURE 3. Concrete crack evaluation from the Mask R-CNN. (a) Ground truth; (b) Model output segmentation mask; (c) Postprocessed segmentation mask.

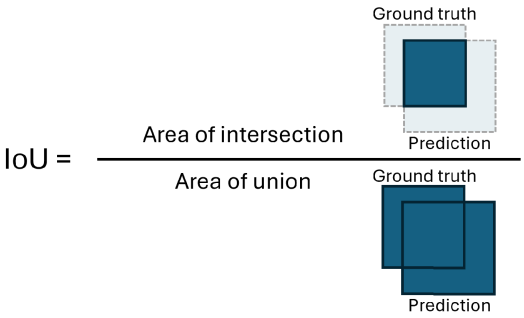

FIGURE 4. Interaction over union (IoU) for the prediction performance evaluation.

III. RESULT AND DISCUSSION

A. PREDICTION PERFORMANCE

Figure 5 shows typical crack detection results from both Mask R-CNN and YOLOv8 models. Both algorithms successfully detect concrete cracks in the majority of test datasets. Quantitatively, Mask R-CNN demonstrates higher accuracy with a mean Intersection over Union (IoU) value of 96.5%, compared to YOLOv8's IoU of 90.6%.

Figure 6 shows the inaccurate detections. Mask R-CNN successfully identifies the crack area, but it also overdetects the non-crack areas as cracks. This false positive detection case is the most common failure mode for Mask R-CNN.

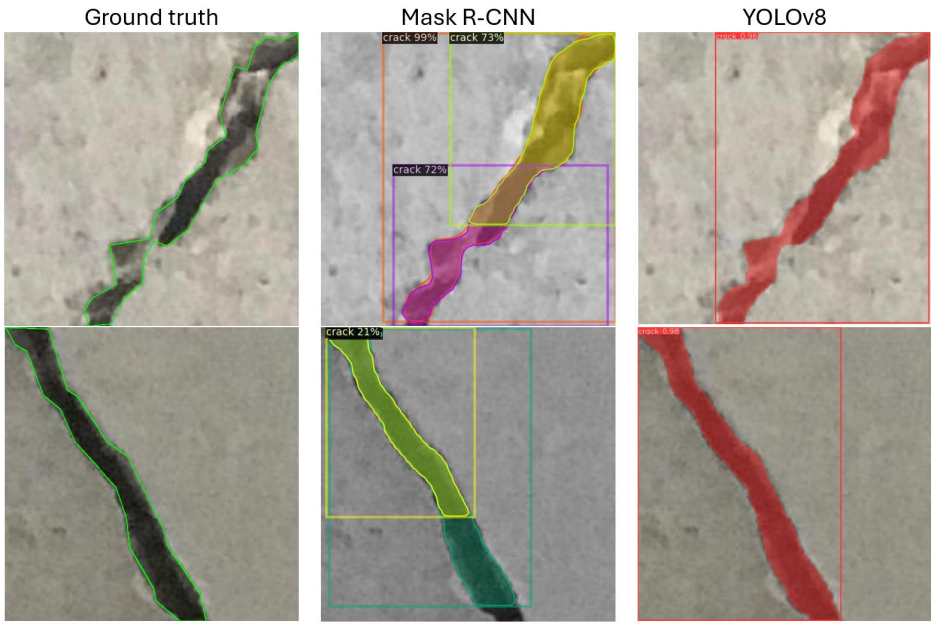

FIGURE 5. Crack detection results from Mask R-CNN and YOLOv8.

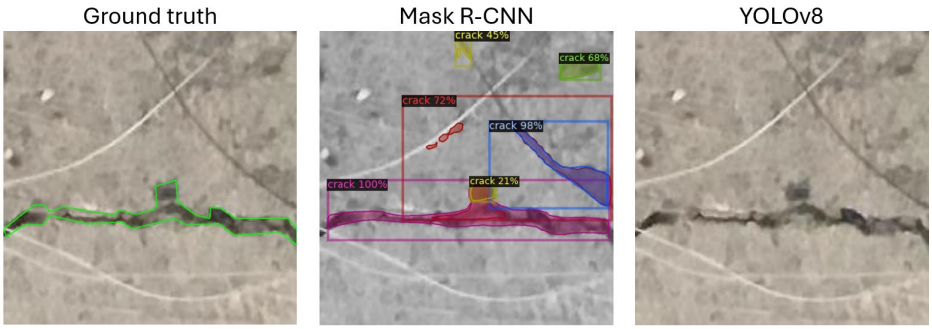

FIGURE 6. Failure case.

In contrast, for the same image, YOLOv8 fails to detect the crack although we can obviously observe the track.

Figure 7 presents the IoU distribution for the test data from YOLOv8 and Mask R-CNN, represented by violin plots. The embedded gray boxes represent box plots, denoting the median and interquartile range (IQR). Individual black dots represent the evaluated IoU datapoints for each test image.

When we look at YOLOv8 and Mask R-CNN with ResNeXt backbone, they both exhibit high performance with IoU values clustered near 100%. The median IoU for both models shows almost the same value with 97.8%. However, Mask R-CNN with ResNeXt demonstrates more consistent detection results, with entire IoU values concentrated above 77% and a narrower IQR. There are no cases where Mask R-CNN completely fails to detect the crack. On the other hand, the YOLOv8's detection is less consistent than Mask R-CNN, and sometimes entirely fails to detect the cracks (IoU of 0%), as discussed in Figure 6. From a structural health monitoring perspective, these false negative cases (failure to detect cracks in a true crack region) are potentially more critical than the false positive cases (Figure 6) that Mask R-CNN tends to produce.

Figure 8 shows examples where the Mask R-CNN successfully detects cracks while YOLOv8 fails. YOLOv8's prediction typically fails by missing some part of the crack area, in contrast to Mask R-CNN's failure, which tends to occur by the overdetection as discussed earlier in Figure 6.

Mask R-CNN's higher accuracy can be attributed to its two-stage process: the region proposal network (RPN) first proposes candidate object regions, which are then refined to generate segmentation masks. The final crack detection segmentation mask is enhanced by one segmentation filling the missing part of the others. Conversely, YOLOv8 employs a single-stage approach where the network directly predicts the bounding boxes and the corresponding segmentation masks. This method occasionally fails to locate the bounding boxes in separate or small instances, resulting in missing segmentation masks.

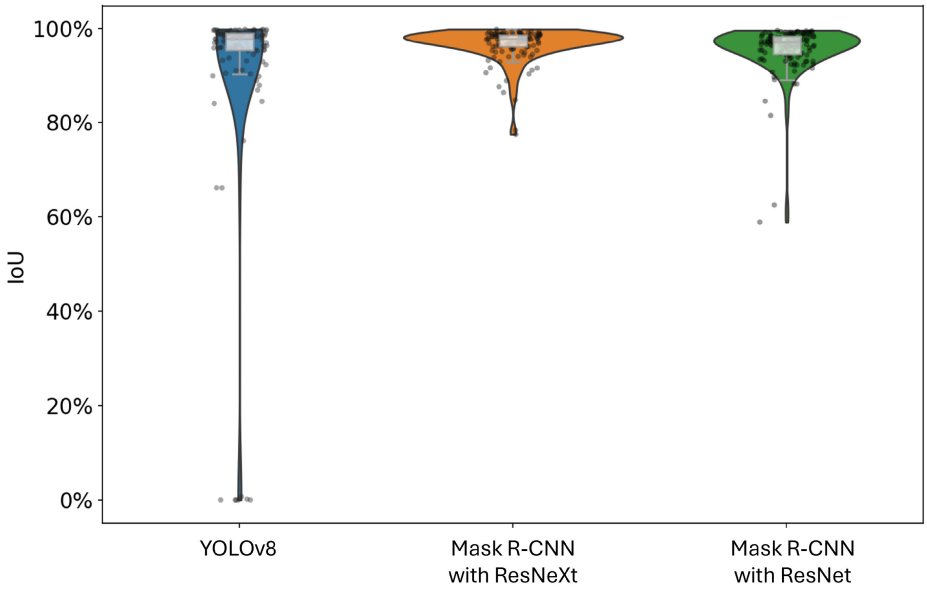

FIGURE 7. Detection performance comparison between YOLO8, Mask R-CNN with ResNeXt, Mask R-CNN with ResNet based on IoU.

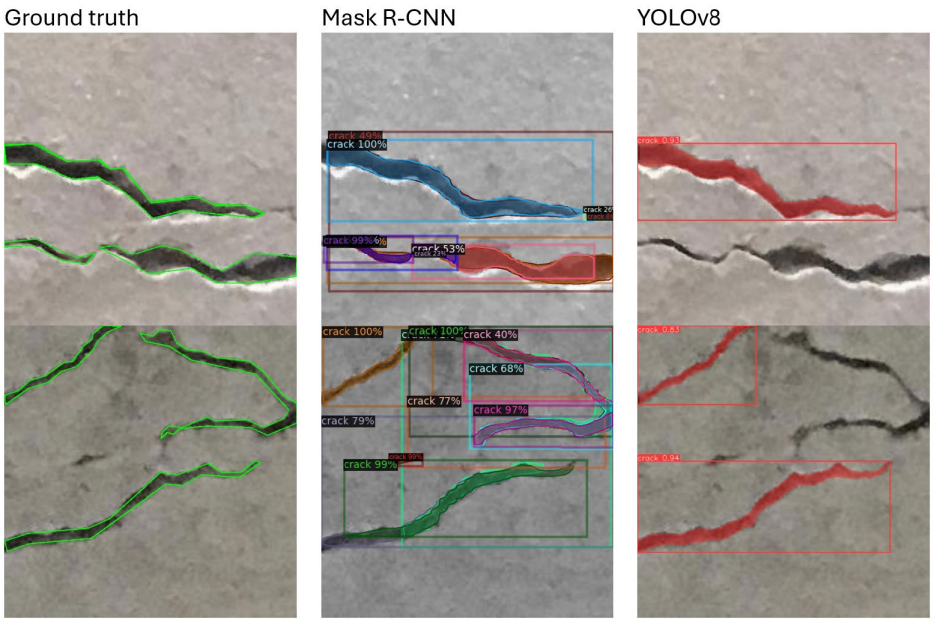

FIGURE 8. Cases where Mask R-CNN shows successful detections, but YOLOv8 does not.

Mask R-CNN model in this study utilizes ResNeXt as the backbone, not like most of the previous studies [17], [18], [23] where their backbone is based on ResNet. This modification yields a performance improvement (see Figure 7). Using the same dataset used in this study, we train and compare ResNet and ResNeXt-based Mask R-CNN models. ResNeXt-based model presents a slight improvement in mean IoU, achieving 96.5% compared to 95.6% of the ResNet-based Mask R-CNN.

Notably, the minimum IoU shows a noticeable improvement with the ResNeXt backbone. The ResNet-based model's minimum IoU is 59%, while the ResNext-based model improves this to 77%. This improvement in minimum IoU suggests that the ResNext backbone provides more consistent performance across diverse crack patterns and image conditions, potentially reducing the occurrence of false detections.

The performance gain can be attributed to the parallel network pathways in ResNeXt within each convolutional layer, which allows the network to learn a richer set of features. The improved feature representation particularly benefits the detection of more diverse

crack instances, as evidenced by the enhanced minimum IoU.

B. PROCESSING TIME AND COMPUTATIONAL EFFICIENCY

In addition to detection accuracy, we evaluated the processing time for both models. Table 1 shows the evaluation time of Mask R-CNN and YOLOv8 for crack detection per image. The computation is conducted on Google Colab T4 GPU. Mask R-CNN, despite its higher accuracy, requires a slightly longer mean detection time of 0.4867 s per image, compared to that of YOLOv8 which requires 0.3225 s per image.

TABLE 1. Mean evaluation time of Mask R-CNN and YOLOv8 per image.

Model	Mean (s)	Standard deviation (s)
Mask R-CNN	0.4867	0.0204
YOLOv8	0.3225	0.0092

This speed difference is attributed to YOLOv8's single-stage architecture, which allows for faster inference by eliminating the separate region proposal step. Although YOLOv8 demonstrates an advantage in processing speed, Mask R-CNN's computation time is not significantly high to hinder the purpose of the task.

Considering the higher accuracy of Mask R-CNN and the non-critical nature of processing speed for most crack detection applications, Mask R-CNN appears to be the more suitable approach. In scenarios where accuracy is prioritized over speed, which is often the case in structural health monitoring and detailed inspections, the performance gain of Mask R-CNN outweighs its slightly longer processing time.

IV. CONCLUSION

This study compares the performance of two state-of-the-art CNN algorithms, Mask R-CNN and YOLOv8, for concrete crack detection. Our results show that both models demonstrate high accuracy in crack detection, with Mask R-CNN outperforming YOLOv8 in terms of detection accuracy with better consistency. The key findings are as follows:

- Among 122 concrete crack images, Mask R-CNN achieves a higher mean Intersection over Union (IoU) of 96.5% compared to YOLOv8's 90.6%, indicating better accuracy in crack detection.
- There are no cases where Mask R-CNN fails to capture the crack, while YOLOv8 sometimes completely misses the crack with the IoU value of 0%. In addition, Mask R-CNN shows more consistent performance, with entire IoU values concentrated above 77%.
- YOLOv8 shows faster processing times, averaging 0.3225 seconds per image compared to Mask R-CNN's 0.4867 seconds.
- For Mask R-CNN, the use of ResNeXt as the backbone results in improved performance compared to the traditional ResNet backbone, particularly in terms of minimum IoU.

Our study provides insights into the concrete crack detection characteristics of the advanced CNN models for instance segmentation. Considering the higher accuracy of Mask R-CNN and the non-critical nature of processing speed for most crack detection tasks, Mask R-CNN appears to be the more suitable approach.

We anticipate that the detection accuracy can be further increased by a larger volume of training data. The improved model can be used for automation and practical application of concrete crack image analysis in the future.

REFERENCES

[1] R. Ali, J. H. Chuah, M. S. A. Talip, N. Mokhtar, and M. A. Shoaib, "Structural crack detection using deep convolutional neural networks," *Autom. Construct.*, vol. 133, Jan. 2022, Art. no. 103989.
[2] Y. Hamishebahar, H. Guan, S. So, and J. Jo, "A comprehensive review of deep learning-based crack detection approaches," *Appl. Sci.*, vol. 12, no. 3, p. 1374, Jan. 2022.
[3] K. C. Laxman, N. Tabassum, L. Ai, C. Cole, and P. Ziehl, "Automated crack detection and crack depth prediction for reinforced concrete structures using deep learning," *Construction Building Mater.*, vol. 370, Mar. 2023, Art. no. 130709.
[4] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: State of the art," *Int. J. Multimedia Inf. Retr.*, vol. 9, no. 3, pp. 171–189, Sep. 2020.
[5] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.
[6] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *Multimedia Tools Appl.*, vol. 82, no. 6, pp. 9243–9275, Mar. 2023.
[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[8] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[9] Q. Wu, Z. Song, H. Chen, Y. Lu, and L. Zhou, "A highway pavement crack identification method based on an improved U-Net model," *Appl. Sci.*, vol. 13, no. 12, p. 7227, Jun. 2023.
[10] Z. Liu, Y. Cao, Y. Wang, and W. Wang, "Computer vision-based concrete crack detection using U-Net fully convolutional networks," *Autom. Construct.*, vol. 104, pp. 129–139, Aug. 2019.
[11] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," *Autom. Construct.*, vol. 99, pp. 52–58, Mar. 2019.
[12] A. Di Benedetto, M. Fiani, and L. M. Gujski, "U-Net-based CNN architecture for road crack segmentation," *Infrastructures*, vol. 8, no. 5, p. 90, May 2023.
[13] L. Ali, F. Alnajjar, H. A. Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures," *Sensors*, vol. 21, no. 5, p. 1688, Mar. 2021.
[14] B. Kim, N. Yuvaraj, K. R. Sri Preethaa, and R. Arun Pandian, "Surface crack detection using deep learning with shallow CNN architecture for enhanced computation," *Neural Comput. Appl.*, vol. 33, no. 15, pp. 9289–9305, Aug. 2021.
[15] V. P. Golding, Z. Gharineiat, H. S. Munawar, and F. Ullah, "Crack detection in concrete structures using deep learning," *Sustainability*, vol. 14, no. 13, p. 8117, Jul. 2022.
[16] S. Teng, Z. Liu, G. Chen, and L. Cheng, "Concrete crack detection based on well-known feature extractor model and the YOLO_v2 network," *Appl. Sci.*, vol. 11, no. 2, p. 813, Jan. 2021.
[17] C. Huang, Y. Zhou, and X. Xie, "Intelligent diagnosis of concrete defects based on improved mask R-CNN," *Appl. Sci.*, vol. 14, no. 10, p. 4148, May 2024.
[18] L. Attard, C. J. Debono, G. Valentino, M. Di Castro, A. Masi, and L. Scibile, "Automatic crack detection using mask R-CNN," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 152–157.
[19] Y. Zhang, J. Huang, and F. Cai, "On bridge surface crack detection based on an improved YOLO v3 algorithm," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 8205–8210, 2020.

[20] Z. Yu, "YOLO V5S-based deep learning approach for concrete cracks detection," in *Proc. SHS Web Conf.*, vol. 144, 2022, p. 03015.
[21] J. Deng, Y. Lu, and V. C.-S. Lee, "Imaging-based crack detection on concrete surfaces using you only look once network," *Struct. Health Monitor.*, vol. 20, no. 2, pp. 484–499, Mar. 2021.
[22] K. Haciefendioğlu and H. B. Başağa, "Concrete road crack detection using deep learning-based faster R-CNN method," *Iranian J. Sci. Technol., Trans. Civil Eng.*, vol. 46, no. 2, pp. 1621–1633, Apr. 2022.
[23] X. Xu, M. Zhao, P. Shi, R. Ren, X. He, X. Wei, and H. Yang, "Crack detection and comparison study based on faster R-CNN and mask R-CNN," *Sensors*, vol. 22, no. 3, p. 1215, Feb. 2022.
[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[25] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5987–5995.
[26] M. Sohan, T. S. Ram, R. Reddy, and C. Venkata, "A review on YOLOv8 and its advancements," in *Proc. Int. Conf. Data Intell. Cognitive Inform.* Singapore: Springer, 2024, pp. 529–545.
[27] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 28, 2015, pp. 1–9.
[28] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[29] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
[30] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1571–1580.
[31] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.
[32] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: A simple and strong anchor-free object detector," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 1922–1933, Apr. 2022.
[33] C. F. Özgenel, "Concrete crack images for classification," Mendeley Data, V2, 2019, doi: 10.17632/5y9wdsg2zt.2.

YONGJIN CHOI received the B.S. and M.S. degrees in civil engineering from Pusan National University, Busan, South Korea, and the Ph.D. degree in civil engineering from The University of Texas at Austin, Austin, TX, USA, in 2024. He is currently a Postdoctoral Fellow with Georgia Institute of Technology. His research interests include accelerating forward and inverse modeling of granular and fluid flows using differentiable machine learning surrogate models, machine learning applications in civil engineering, and material point methods for large deformation hazard analysis.

BYONGKYU BAE received the B.S. degree in civil engineering from Pusan National University, Busan, South Korea, in 2023, where he is currently pursuing the M.S. degree in civil and environmental engineering. His research interests include computer vision with machine learning and its application to civil infrastructures.

TAEK HEE HAN received the B.S. degree in civil engineering and the M.S. and Ph.D. degrees in structural engineering from Korea University, in 1995, 2001, and 2006, respectively. He was with Auburn University and Seoul Metro and has spent 14 years with Korea Institute of Ocean Science and Technology, focusing on high-performance marine and port structures and marine energy development. He is currently leading the research and development of underwater space platforms and habitation modules for human living in underwater environments as well as underwater data centers aimed at enhancing carbon reduction and energy efficiency.

JAEHUN AHN received the M.S. degree in tunneling from Korea University, and the Ph.D. degree in soil dynamic properties from Texas A&M University. After leaving Texas A&M University, he conducted research on subgrade materials with North Carolina State University, granular materials for water filtering with the University of Connecticut, offshore foundations, geotechnical systems, and permeable pavements with Pusan National University based on the application of soil dynamics, numerical and experimental modeling, deep learning, and sensors and data acquisition. He is currently a Professor with Pusan National University, South Korea.