applied sciences

MDPI

Article

CL-YOLOv8: Crack Detection Algorithm for Fair-Faced Walls Based on Deep Learning

Qinjun Li, Guoyu Zhang and Ping Yang *

School of Information Engineering & Artificial Intelligence, Shaanxi University of Science & Technology, Xi'an 710026, China; liqinjun@sust.edu.cn (Q.L.); 221612132@sust.edu.cn (G.Z.)
* Correspondence: yangping@sust.edu.cn

Abstract: Cracks pose a critical challenge in the preservation of historical buildings worldwide, particularly in fair-faced walls, where timely and accurate detection is essential to prevent further degradation. Traditional image processing methods have proven inadequate for effectively detecting building cracks. Despite global advancements in deep learning, crack detection under diverse environmental and lighting conditions remains a significant technical hurdle, as highlighted by recent international studies. To address this challenge, we propose an enhanced crack detection algorithm, CL-YOLOv8 (ConvNeXt V2-LSKA-YOLOv8). By integrating the well-established ConvNeXt V2 model as the backbone network into YOLOv8, the algorithm benefits from advanced feature extraction techniques, leading to a superior detection accuracy. This choice leverages ConvNeXt V2's recognized strengths, providing a robust foundation for improving the overall model performance. Additionally, by introducing the LSKA (Large Separable Kernel Attention) mechanism into the SPPF structure, the feature receptive field is enlarged and feature correlations are strengthened, further enhancing crack detection accuracy in diverse environments. This study also contributes to the field by significantly expanding the dataset for fair-faced wall crack detection, increasing its size sevenfold through data augmentation and the inclusion of additional data. Our experimental results demonstrate that CL-YOLOv8 outperforms mainstream algorithms such as Faster R-CNN, YOLOv5s, YOLOv7-tiny, SSD, and various YOLOv8n/s/m/l/x models. CL-YOLOv8 achieves an accuracy of 85.3%, a recall rate of 83.2%, and a mean average precision (mAP) of 83.7%. Compared to the YOLOv8n base model, CL-YOLOv8 shows improvements of 0.9%, 2.3%, and 3.9% in accuracy, recall rate, and mAP, respectively. These results underscore the effectiveness and superiority of CL-YOLOv8 in crack detection, positioning it as a valuable tool in the global effort to preserve architectural heritage.

Keywords: ConvNeXt V2; crack detection; fair-faced wall; LSKA; YOLOv8

check for updates

Citation: Li, Q.; Zhang, G.; Yang, P. CL-YOLOv8: Crack Detection Algorithm for Fair-Faced Walls Based on Deep Learning. *Appl. Sci.* **2024**, *14*, 9421. https://doi.org/10.3390/app14209421

Academic Editor: Andrea Prati

Received: 20 September 2024
Revised: 9 October 2024
Accepted: 11 October 2024
Published: 16 October 2024

Copyright: © 2024 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

Many historical buildings are marked by walls constructed using plain bricks, known as "fair-faced walls", which directly showcase the texture and color of these bricks without additional decorative layers. However, due to factors such as the natural environment, material aging, and human interference, these walls often suffer from ailments like cracks, posing serious threats to the safety and integrity of historic buildings. Cracks compromise the aesthetic appeal of historical buildings and risk further structural deterioration, potentially leading to catastrophic consequences such as collapse. For example, the city wall ruins of the Ming and Qin Dynasties in Xi'an, Shaanxi Province, China, collapsed due to cracks caused by rainwater erosion. Similarly, cracks were contained in the walls of the Yangjialing Revolutionary Site Museum in Yan'an, Shaanxi Province, China. Therefore, the timely and accurate detection and assessment of these cracks are of the utmost importance for the protection and maintenance of historical buildings.

Traditional crack detection methods mainly rely on manual on-site surveys and recordings, which are inefficient and susceptible to subjective influences, making it difficult to

meet the protection needs of large-scale historical buildings. While some simple image processing techniques offer convenience compared to manual inspections, they still suffer from issues such as missed detections and false alarms.

With the advancement of technology, various techniques beyond artificial intelligence have also played significant roles in this field. For example, Wojciechowska, G. et al. [1]. integrated intelligent monitoring systems with digital image processing technology for building monitoring and maintenance. By using high-resolution cameras and image processing algorithms, cracks can be detected early and their progression can be monitored. Digital image processing techniques, such as filtering, edge detection, and threshold segmentation, are employed to extract the geometric features of cracks. While these methods perform well in relatively uniform environments, their robustness under complex lighting and environmental conditions is not as strong as that of deep learning techniques. Nonetheless, these traditional methods remain fundamental in building monitoring, especially for simpler crack detection tasks. Golewski et al. [2]. discussed the application of material modification in building structures. By using fly ash (FA) and nano-silica (NS), the crack resistance of materials can be significantly improved. This approach enhances the mechanical properties of materials, reducing the formation of cracks and addressing potential issues in the design and construction stages of building structures. This type of material science technology serves as a preventive measure for cracks and can complement modern detection techniques. In addition to image-based detection methods, modern crack detection also includes ultrasonic testing, infrared thermography, and laser scanning technology. These non-AI methods each offer advantages in crack detection for building structures. For instance, digital image processing techniques remain a reliable foundational approach, while material enhancement technologies help to reduce the need for subsequent crack detection by preventing their occurrence. By combining these methods with artificial intelligence, more comprehensive and precise monitoring and crack management strategies can be formulated.

In recent years, with the rapid development of artificial intelligence and deep learning, significant progress has been achieved in wall defect detection using convolutional neural network (CNN)-based object detection algorithms. Based on the different processes of candidate region generation and recognition during detection, object detection algorithms can be classified into the two following main types: two-stage detection algorithms and one-stage detection algorithms. Among two-stage detection algorithms, common examples include the R-CNN series (R-CNN, Fast R-CNN, and Faster R-CNN) and Mask R-CNN [3], which achieve object detection by first generating candidate regions and then recognizing them. Borja Marin et al. proposed a detection method for cracks in masonry structures using the architecture of the Faster R-CNN object detector and the Mobilenetv2 network [4]. They introduced a new approach called "progressive detection" to enhance the robustness of detection. Chen et al. proposed a method for detecting cracks in building facades using unmanned aerial vehicles (UAVs) and computer vision technology. They selected Mask R-CNN, which demonstrated the best performance, as the detection network, achieving an average precision of 0.9 [5]. Although two-stage detection algorithms demonstrate an exceptional accuracy in detecting targets in complex scenes, they need to first generate candidate regions, extract features, and then recognize each candidate region, leading to relatively slow detection speeds. This limitation makes it challenging to respond to real-time requirements, and the generation of candidate regions and subsequent feature extraction involve a certain redundancy and repetitive computations, thereby reducing algorithm efficiency.

One-stage algorithms eliminate the step of candidate region generation and directly perform feature extraction and object detection on the entire image, significantly simplifying the detection process and improving the detection speed. The YOLO (You Only Look Once) series algorithms represent one-stage detection algorithms, including YOLOv1, YOLOv2, YOLOv3, YOLOv4, and YOLOv5. The YOLO algorithm divides an image into grids, with each grid being relied on to predict targets within its region. By simultaneously

performing feature extraction, target classification, and position regression, the YOLO algorithm achieves real-time object detection, dramatically enhancing detection speed while maintaining detection accuracy. Additionally, the SSD (Single-Shot MultiBox Detector) series algorithms and OverFeat are also designed based on the one-stage concept, continuing to improve the performance of one-stage algorithms through different network structures and optimization strategies. Pratibha K et al. deployed an automated strategy based on the YOLOv5 object detection model for crack detection in brick structures, enabling the precise localization of cracks through bounding box capture. The model achieved real-time health monitoring capabilities [6]. Su et al. designed a YOLO algorithm, named MOD-YOLO [7], which preserves the original contour. By introducing depth-wise separable convolution and Discriminative and Average Fusion Coordinated Attention (DAF-CA), they improved the YOLOX algorithm and applied it to crack detection in civil infrastructure. Nabizadeh E et al. considered the question of surface crack detection in concrete structures by comparing the detection capabilities of the YOLOv8 and YOLOv5 models. They concluded that the YOLOv8 algorithm demonstrated a superior performance in terms of detection accuracy [8]. YOLOv8 is a new model in the YOLO series, known for its high detection accuracy, smaller parameter count, and model size.

Currently, most deep-learning-based crack detection methods are primarily applied to modern architectural structures, such as pavement crack detection, bridge crack detection, and concrete structure crack detection. However, historical buildings constructed with fair-faced wall structures present unique challenges. These structures not only develop cracks due to their age, but the types of cracks found in these walls differ significantly from those found in other types of structures. The cracks in fair-faced walls are typically wider and pose greater risks. Therefore, this study focuses on addressing the specific challenges of crack detection in fair-faced wall structures, which is of significant importance for the preservation of historical buildings.

In this study, we addressed the challenges of ineffective crack detection on fair-faced walls, particularly due to the varying crack characteristics and low detection accuracy across different environmental conditions. We proposed an improved crack detection algorithm based on YOLOv8. The main contributions of this research are summarized as follows:

1. We reconstructed the backbone network using the ConvNeXt V2 model. This redesigned network demonstrated a superior detection performance, especially on wall surfaces with less prominent crack features.
2. The introduction of the LSKA attention mechanism within the Spatial Pyramid Pooling-Fast (SPPF) structure enhanced the model's focus on cracks, leading to an improved detection accuracy across various environmental scenarios.
3. To address the limitations of the existing open-source dataset, we supplemented it with additional data and employed data annotation and augmentation techniques, significantly expanding and diversifying the dataset.

2. Related Work

2.1. Methods in Computer Vision

Convolutional Neural Networks (CNNs) and the recently introduced Vision Transformers (ViTs) have achieved significant advancements in several key computer vision tasks, including object recognition, detection, and segmentation. For instance, Yang et al. proposed the Modality Fusion Vision Transformer (MFViT) [9], a multimodal data fusion framework designed for the classification of hyperspectral imaging (HSI) and LiDAR data. This model features a novel modality fusion module that enhances the efficiency of HSI feature extraction by focusing the spatial characteristics of LiDAR data onto HSI data. Although the MFViT demonstrates some advantages in terms of parameter efficiency, its computational complexity surpasses that of other Transformer-based models. Zheng et al. applied Vision Transformers to the domain of traffic sign recognition and found that, while Transformers can outperform CNNs in digital holography (DH) tasks, CNNs remain

more competitive in traffic sign classification [10]. This study suggests that combining convolutional operations with attention mechanisms can help to bridge the performance gap between CNNs and Transformers. Chen et al. introduced a dual-branch Transformer architecture designed to generate stronger image features [11]. This model processes small and large patch tokens through branches with different computational complexities and then fuses these tokens multiple times using attention mechanisms to achieve complementary results. This approach performs on par with or better than several current Vision Transformers, but still falls slightly short when compared to CNN models. Simone Ercolino and colleagues demonstrated that fine-tuned ViT-based models [12], such as the Dense Prediction Transformer (DPT), offer a state-of-the-art performance in depth estimation from drone imagery, surpassing fully convolutional neural networks in both accuracy and robustness. However, due to their self-attention operations, Vision Transformers entail higher computational costs than CNNs, resulting in slower processing speeds on edge devices in real-world applications.

2.2. Feature Extraction

In computer vision tasks, feature extraction plays a key role in extracting useful information from input images through neural networks and converting them into a series of feature representations. In YOLOv8, the feature extraction network transforms the input image into a feature map containing various image features such as color, texture, edges, etc. This feature map is then fed into the detection network to recognize objects in the image and predict their positions and categories. Therefore, the feature extraction network plays a vital role in deep learning, determining whether the network can extract useful information from the input data to accomplish tasks.

In the task of wall crack detection, wall cracks come in numerous shapes and sizes, including horizontal, vertical, and fractured cracks. Moreover, the appearance of wall cracks can vary in different environmental conditions, where changes in lighting conditions on the wall surface due to factors such as time and weather can lead to variations in crack illumination and shadows. Therefore, increasing the feature extraction capability in wall crack detection tasks can address challenges such as crack diversity, environmental changes, and background interference, thereby improving detection accuracy and robustness.

Several studies have made improvements in feature extraction networks. For instance, Xu et al. proposed an approach that adds two feature extraction modules (DPFE and IRFE) [13] to the feature extraction network and detection layer while maintaining a lightweight backbone network, achieving iterative feature extraction. Chen et al. introduced an improvement based on attention mechanisms, enhancing the feature extraction capability by adding an Extended Attention Module (DAM) [14] to extract salient features from different branches. Ran et al. proposed a Multi-Scale Context and Enhanced Channel Attention (MSCCA) model [15], which improves the representation of object features in complex backgrounds and enhances the representation capability of convolutional neural networks. Zhou et al. utilized the ConvNeXt model [16] to construct a multi-scale feature extraction network for the YOLOv5 model, enhancing its detection capability for low-resolution infrared images.

2.3. Attention Mechanism

In deep learning, attention mechanisms are often added to the model architecture to enhance its performance by increasing the focus and weight allocation of neural networks on input images. These mechanisms dynamically adjust the attention of various parts within the network based on the importance of input data, allowing the network to concentrate more on meaningful information for the current task, thus enhancing its performance. The mainstream attention mechanisms include spatial attention, channel attention, multi-head attention, and self-attention. Zhang et al. proposed adding a CBAM (Convolutional Block Attention Module) [17] in the feature extraction stage, consisting of channel attention and spatial attention sub-modules, which would improve the network's detection performance

for multi-scale targets. Shen et al. introduced a dual-route attention mechanism between the backbone network and the neck network, effectively combining spatial and channel information to enhance the model's feature learning capability [18]. Liu et al. introduced Large Kernel Attention (LKA) [19] into CNNs, decomposing convolution operations to expand the model's receptive field and enhance its feature extraction capability. The LKA mechanism differs from traditional attention mechanisms based on small convolutional kernels by using larger convolutional kernels to calculate attention weights. Leveraging large kernel convolutions enables capturing a broader range of feature information and more contextual information, thereby enhancing the model's feature representation capability.

3. Proposed Methods

3.1. Overview of YOLOv8

YOLOv8 is the latest YOLO model, launched by Ultralytics in January 2023. YOLOv8 is divided into the five following model networks: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. In this article, we selected the YOLOv8n network, which offers a smaller size and higher accuracy.

The YOLOv8n network consists primarily of the four following main components: the input, backbone, neck, and head, as depicted in Figure 1.

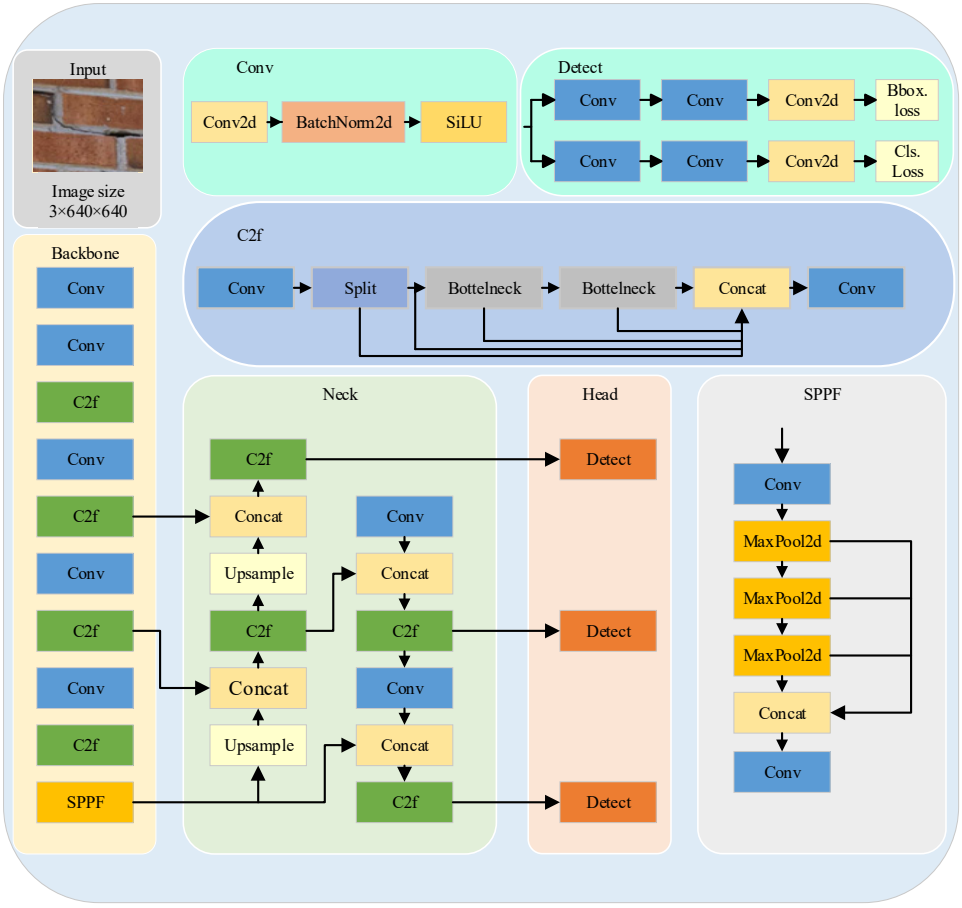

Figure 1. The system diagram of YOLOv8n.

In the input stage, the YOLOv8 algorithm employs methods such as Mosaic data augmentation and adaptive anchor box calculation to enrich the dataset, enhancing the model's generalization ability and robustness [20]. The backbone network consists of modules such as Conv, C2f, and SPPF, which are employed for feature extraction. YOLOv8n introduces a new C2f structure that not only contains rich gradient flow information, but also enhances the feature fusion capability of the CNN network [21]. The SPPF module (Spatial Pyramid

Pooling) can transform feature maps of any size into fixed-size feature vectors, enabling the fusion of local and global features [22]. The neck network is responsible for multiscale feature fusion, forming a feature pyramid. The neck section consists of the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN) structures. YOLOv8 replaces the C3 module in the FPN-PAN with the C2f module, while the other components remain largely consistent with the FPN-PAN structure. In the output stage, YOLOv8n adopts the anchor-free approach, replacing the traditional anchor-based method. This is a significant change that shifts the detection head from a "coupled" structure in YOLOv3 to a mainstream "decoupled head" structure in YOLOv5. It separates the classification and detection heads into two parallel branches to perform classification and localization tasks. Overall, YOLOv8 has a simpler network structure compared to YOLOv5, while achieving a higher accuracy and faster detection speeds.

3.2. Introducing the LSKA Mechanism Module

The attention mechanism is typically used to enhance the focus of a model on specific parts or features of input data. In the context of object detection tasks, utilizing attention mechanisms can help the model to concentrate on the relevant features of the detected objects, thereby enhancing its detection accuracy. Visual Attention Networks (VANs) with Large Kernel Attention (LKA) modules have been proven to offer an excellent performance across a range of visual tasks. Large kernels refer to convolutional kernels with larger receptive fields, capable of capturing more extensive contextual information, albeit at a higher computational cost. However, in scenarios involving large kernels, allocating weights to features at each position may lead to efficiency issues. For instance, in the LKA module, the depth convolutional layer's computational and memory usage increase geometrically with the kernel size.

The Large Separable Kernel Attention (LSKA) module deals with this issue by decomposing traditional 2D convolutional kernels into two 1D kernels. Firstly, it decomposes a large 2D kernel into horizontal and vertical 1D kernels, significantly reducing the parameter count and computational complexity. During the convolution operation, LSKA first performs horizontal convolution on the input using one 1D kernel, followed by vertical convolution using another 1D kernel. These consecutive convolution operations achieve similar effects to the original large 2D kernel [23]. By innovatively decomposing kernels and employing consecutive convolution, LSKA maintains the advantages of utilizing large kernels while reducing computational and memory costs. A comparison between LKA and LSKA modules is illustrated in Figure 2.

A large kernel convolution can be divided into the three following components: a spatial depth-wise convolution, a depth-wise dilation convolution, and a channel convolution (1×1 convolution) [24]. Following the methodology outlined in reference [23], the output of LSKA can be obtained as follows.

$$\overline{Z}^C = \sum_{H,W} W^C_{(2d-1)\times 1} * \left(\sum_{H,W} W^C_{1\times(2d-1)} * F^C \right) \tag{1}$$

$$Z^C = \sum_{H,W} W^C_{\lfloor \frac{k}{d} \rfloor \times 1} * \left(\sum_{H,W} W^C_{1\times \lfloor \frac{k}{d} \rfloor} * \overline{Z}^C \right) \tag{2}$$

$$A^C = W_{1\times 1} * \overline{Z}^C \tag{3}$$

$$\overline{F}^C = A^C \otimes F^C \tag{4}$$

where d is the dilation rate. $\overline{Z}^C$ in Equation (1) denotes the output of depth-wise convolution with a kernel size $(2d-1) \times 1$, which captures the local spatial information and compensates for the gridding effect [24] of the following depth-wise dilated convolution in Equation (2) with a kernel size $\left(\left\lfloor \frac{k}{d} \right\rfloor \times 1 \right)$. Note that $\lfloor \cdot \rfloor$ represents the floor operation. The dilated

depth-wise convolution is responsible for capturing the global spatial information of the output $\overline{Z}^C$ of the depth-wise convolution. The LSKA model is shown in Figure 3.

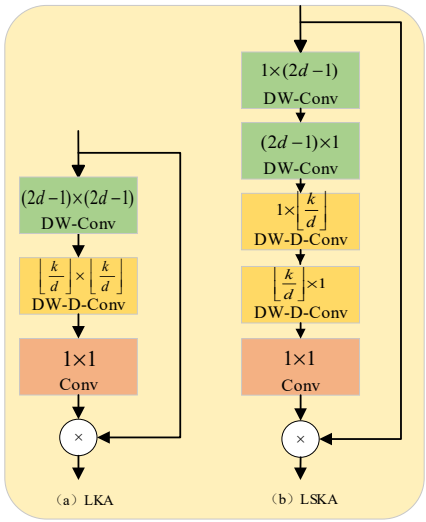

Figure 2. System diagram comparing LKA and LSKA modules.

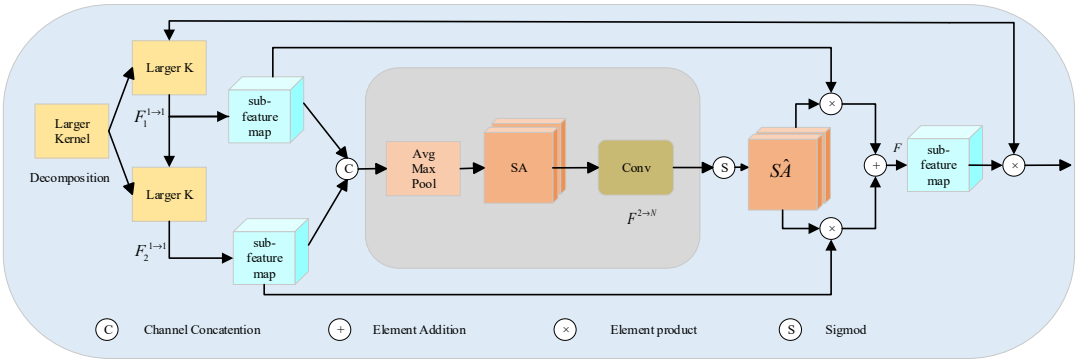

Figure 3. LSKA module.

YOLOv8 introduces the Spatial Pyramid Pooling-Fast (SPPF) module, an optimized and improved version of the traditional Spatial Pyramid Pooling (SPP) module [25]. SPPF significantly reduces the computational complexity by adopting efficient pooling strategies, including reducing the number of pooling layers and merging multiple pooling operations [26]. As LSKA effectively integrates multi-scale features, it ensures the proper reflection of the critical information at each scale and position. By utilizing separable kernels, it highlights the crucial features at different scales and positions with a higher precision. Therefore, integrating the LSKA module into YOLOv8's SPPF structure further enhances its spatial awareness and feature extraction capabilities. Moreover, decomposing large kernels into smaller separable kernels enables LSKA to expand the receptive field to capture rich contextual information while reducing computational costs, thus improving the detection efficiency. The structure of LSKA is depicted in Figure 4.

3.3. Introducing the ConvNeXt V2 Module

The ConvNeXt model is proposed by leveraging the network structure of the Swin Transformer and utilizing the ResNet-50 architecture as a foundation. The performance of the ConvNeXt model surpasses that of the Swin Transformer in COCO detection and ADE20K tasks. The authors of the ConvNeXt model employed strategies from Vision Transformers (ViTs) to train the ResNet-50 network, resulting in superior results compared to the original model [27].

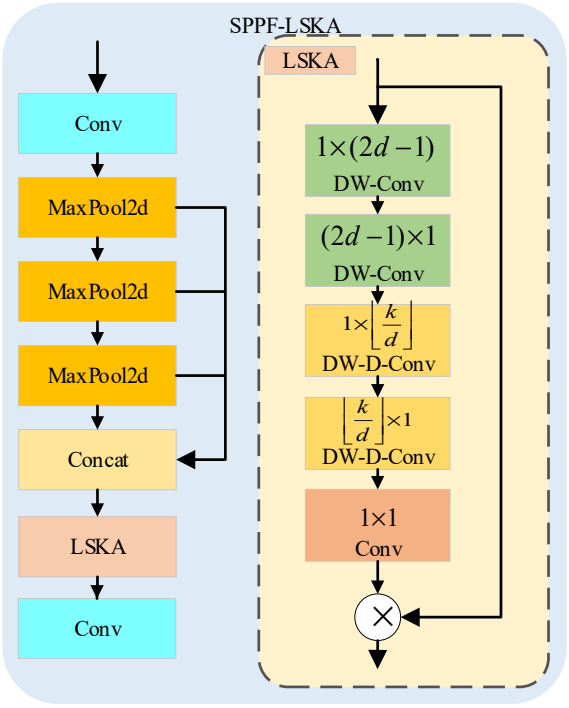

Figure 4. Diagram of the LSKA module structure.

The ConvNeXt V2 model is a convolutional neural network derived from the ConvNeXt model, incorporating self-supervised learning techniques and architectural improvements. It integrates the Fully Convolutional Mask Autoencoder (FCMAE) and Global Response Normalization (GRN) layers to enhance the feature competition among channels. Through methods such as global feature aggregation, feature normalization, and feature calibration, ConvNeXt V2 strengthens the inter-channel feature competition [28], thereby increasing the contrast and selectivity among channels. This enhancement improves the feature extraction capability of the backbone network on complex images. A comparison between ConvNeXt V2 Block and ConvNeXt V1 Block is illustrated in Figure 5.

Compared to ConvNeXt V1, ConvNeXt V2 adds a Global Response Normalization (GRN) layer after the Multi-Layer Perceptron (MLP) layer and removes the redundant LayerScale in ConvNeXt V2. The GRN layer initially takes an input feature $X \in R^{H\times W\times C}$ and aggregates spatial feature maps X_i into a vector gx with a global function $G(\cdot)$, as shown in Equation (5). Next, standard division normalization is applied, where the normalization function $\mathcal{N}(\cdot)$ operates on the aggregated value, as depicted in Equation (6). Finally, the computed normalized scores are used to calibrate the original input responses along with the original spatial feature maps, as illustrated in Equation (7) [28,29].

$$G(x) := X \in R^{H\times W\times C} \rightarrow gx \in R^{C} \tag{5}$$

$$\mathcal{N}(\|X_i\|) := \|X_i\| \in R \rightarrow \frac{\|X_i\|}{\sum\limits_{j=1,\ldots,C}\|X_j\|} \in R \tag{6}$$

$$X_i = X_i * \mathcal{N}(G(X)_i) \in R^{H\times W} \tag{7}$$

Replacing the backbone feature extraction network in YOLOv8 with the ConvNeXt V2 model helps to improve YOLOv8's capability for extracting features from complex images. This enhancement contributes to an overall improvement in detection accuracy and performance.

The ConvNeXt V2 model better understands the high-level structure of the data and extracts more meaningful feature representations, leading to an improved performance in image detection tasks [30]. This is because the ConvNeXt V2 model consists of both depth-

wise convolution and point-wise convolution [31]. The primary advantage of depth-wise convolution is the reduction in the parameter count and computational complexity, thus improving the model efficiency. Point-wise convolution reduces the number of channels in depth-wise convolution layers, decreasing the overall computational cost of the network. Additionally, point-wise convolution aids in increasing the feature interaction among channels, thereby enhancing the richness and expressive power of the model's feature representation. The final network model is depicted in Figure 6.

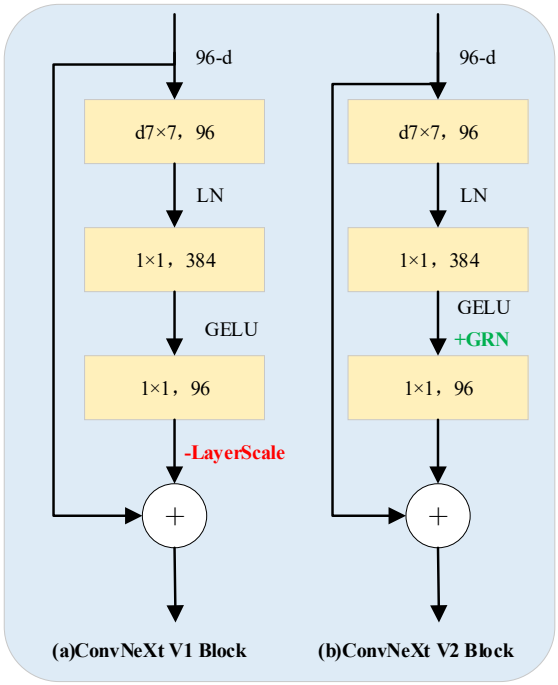

Figure 5. Comparison diagram of ConvNeXt V1 and V2 modules.

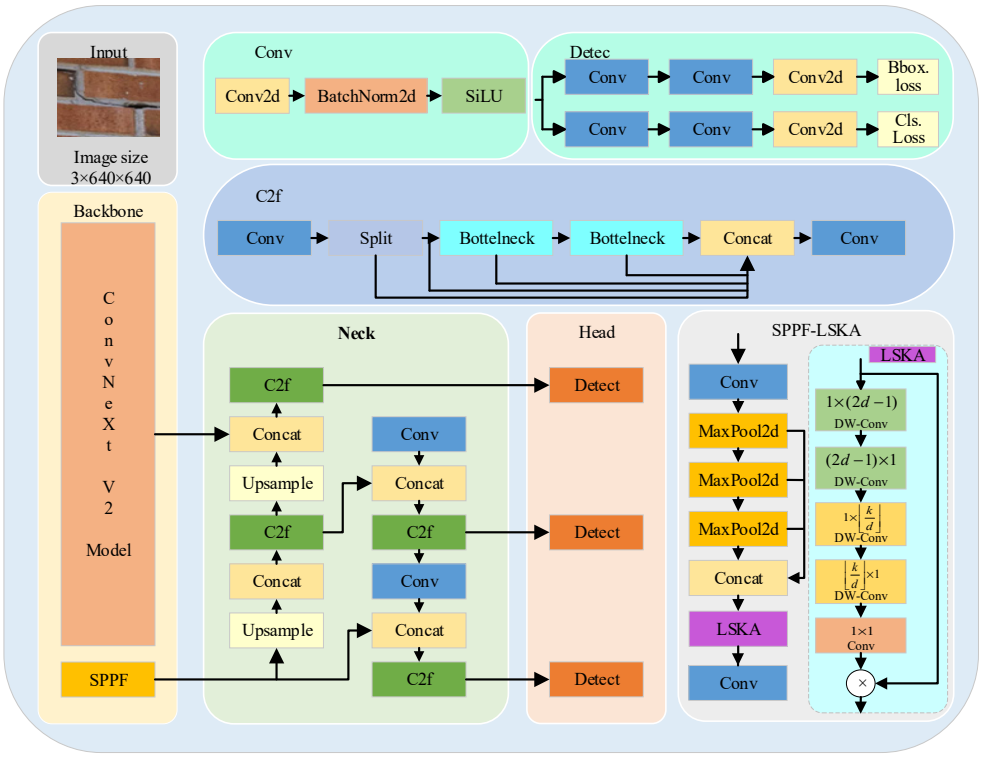

Figure 6. Improved system architecture diagram.

4. Experiments

4.1. Experimental Setup and Dataset

The experimental setup for this study includes a system running on Windows 10 with an Intel Core i5-13400F CPU (Intel Corporation, Santa Clara, CA, USA) and an NVIDIA GeForce RTX 3060 Ti GPU (NVIDIA Corporation, Santa Clara, CA, USA) with 8 GB of VRAM. The development environment is CUDA 11.8, and PyTorch 2.1.2 is used as the deep learning framework for training. The hyperparameters are configured with a batch size of 16 and a training period of 300 epochs.

The dataset for this experiment is an augmentation of the Brickwork Cracks Dataset provided in [32]. The original dataset comprises 350 images of brickwork with and without cracks, each at a resolution of 227 × 227 pixels. To meet the input requirements of the YOLOv8n model, this dataset is preprocessed using the Roboflow tool, which automatically orients and resizes the images to 640 × 640 pixels. Subsequently, data augmentation techniques such as horizontal flipping, 90° rotation, random rotation, and random brightness adjustment are applied to augment the dataset to 5202 images, annotated into the following two categories: crack and no crack. Among these, there are 309 images in the test set, 4584 images in the training set, and 309 images in the validation set, all of which were randomly assigned.

4.2. Evaluation Metrics

To effectively demonstrate the research outcomes, experiments were conducted to evaluate the improved YOLOv8n algorithm. Evaluation metrics such as Precision (P), Recall (R), F1-score, Average Precision (AP), and Mean Average Precision (mAP) were employed to assess the model [31]. Precision refers to the ratio of true positives (TPs) to all samples classified as positive (TPs + FPs) in the detection results. Its calculation formula is as follows:

$$P = \frac{TP}{TP + FP} \tag{8}$$

Recall refers to the ratio of true positives to all true positives in the detection results. Its calculation formula is as follows:

$$R = \frac{TP}{TP + FN} \tag{9}$$

where TP represents the number of true positives correctly classified by the classifier, FP represents the number of false positives incorrectly classified by the classifier, TN represents the number of true negatives correctly classified by the classifier, and FN represents the number of false negatives incorrectly classified by the classifier [33].

The $F1$-score is the harmonic mean of precision (P) and recall (R), with values ranging from 0 to 1. A score closer to 1 indicates a better performance of the algorithm [34]. Its calculation formula is as follows:

$$F1 = \frac{2 \times P \times R}{P + R} \tag{10}$$

The Average Precision (AP) represents the average classification performance for a single category. It is the area under the Precision–Recall (P–R) curve. A higher AP value indicates a higher detection accuracy for that category. Its calculation formula is as follows:

$$AP = \int_0^1 P(R)dR \tag{11}$$

The Mean Average Precision (mAP) is computed by averaging the AP values calculated for all categories. It serves as an evaluation metric for the overall performance of the model. Its calculation formula is as follows:

$$mAP = \frac{1}{n}\sum_{i=1}^{n}(AP)_i \tag{12}$$

In the formula, i represents the total number of categories, mAP50 denotes the overall average precision at an IoU threshold of 0.5, and mAP50:95 represents the average precision across different IoU thresholds (ranging from 0.5 to 0.95, with a step size of 0.05).

4.3. Improved Result

After training the improved CL-YOLOv8 model on a self-curated crack dataset and testing it on 309 images, significant performance enhancements were evident. The training results revealed an accuracy of 85.3%, a recall rate of 83.2%, and a mean average precision (mAP) of 83.7%. As evidenced by the precision–recall (P–R) curve, compared to the original model, both the accuracy and mAP values for crack detection saw improvements. The F1-confidence curve graph demonstrated a notable increase in detection confidence for both the crack and no crack categories. This suggests that, by introducing the LSKA attention module and replacing the backbone feature extraction network, the CL-YOLOv8 model exhibited a higher precision and feature extraction capability in the task of detecting cracks in fair-faced walls. The results are illustrated in Figures 7 and 8.

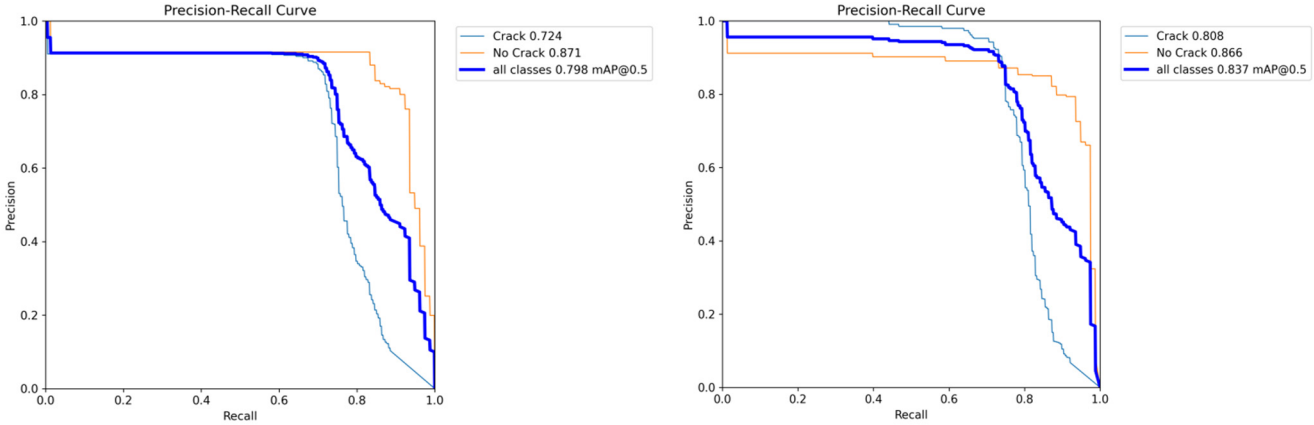

Figure 7. Comparison of P–R figures before and after improvement.

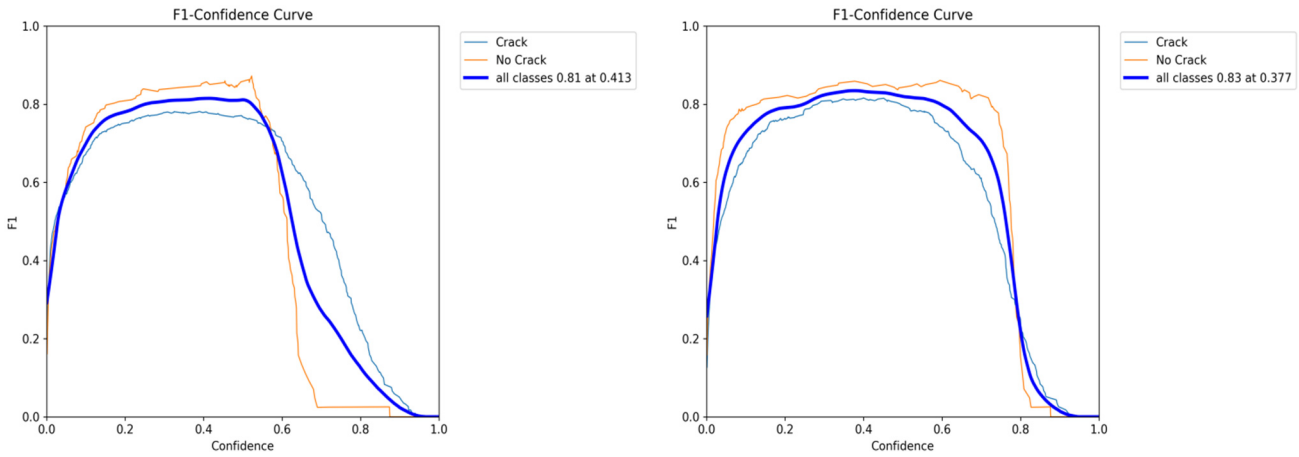

Figure 8. Comparison of F1-confidence curves before and after improvement.

4.4. Contrast Experiment

To comprehensively validate the detection performance of the improved YOLOv8 algorithm proposed in this paper, we conducted comparisons with current state-of-the-art models for wall crack detection under the same configuration environment, using identical validation and training datasets. We selected Precision, Recall, mAP@0.5, mAP@0.5:0.95, Parameters, and GFLOPs (giga floating-point operations per second) as our evaluation metrics. The improved algorithm was compared against the five following models: YOLOv8,

Faster R-CNN, YOLOv5s, and YOLOv7-tiny. The comparative results are summarized in Table 1.

Table 1. Performance comparison of different models.

Model	P/%	R/%	mAP@0.5/%	mAP@0.5:0.95/%	Parameters/10^6	GFLOPs
Faster R-CNN	79.4	78.3	77.5	45.8	28.39	109.65
SSD	73.4	65.9	70.4	36.5	26.3	52.7
YOLOv5s	88.7	75.8	80.7	45.5	7.02	15.8
YOLOv7-tiny	76.3	69	67.9	32.8	6.23	13.9
YOLOv8s	80.4	78.4	78.6	47.2	11.1	28.6
YOLOv8m	85.1	80.3	80.9	48.8	25.9	79.0
YOLOv8l	82.8	77.3	79.3	48.9	43.6	165.3
YOLOv8x	86.4	77.3	80.4	48.7	68.2	258
YOLOv8n	84.4	80.9	79.8	48.5	3.0	8.1
CL-YOLOv8	85.3	83.2	83.7	46.6	6.0	14.4

According to Table 1, YOLOv8n exhibited higher detection precision, recall rate, and mAP values compared to the Faster R-CNN, YOLOv7-tiny, YOLOv8s, and YOLOv8l models. Although YOLOv5s slightly surpasseed YOLOv8n in some metrics, the parameters and total floating-point operations of YOLOv5s were approximately twice those of YOLOv8n. Moreover, YOLOv8n had fewer parameters and GFLOPs compared to the other seven networks, making it the chosen base model for improvement. The proposed CL-YOLOv8 algorithm, while slightly lower in accuracy compared to YOLOv5s and YOLOv8x, had fewer parameters and computational demands than these models. Additionally, its recall rate and mAP values exceeded those of YOLOv5s and YOLOv8x. CL-YOLOv8 outperformed in terms of recall rate and mAP values as well. Despite doubling the parameter count and total floating-point operations compared to the original YOLOv8n, the improved model showed enhanced precision, recall rate, and mAP values, with a 3.9% increase in mAP compared to the original YOLOv8n algorithm.

At the same time, this study compared the ConvNeXt V2 model with other advanced feature extraction networks and evaluated the LSKA attention mechanism against other attention modules. The results demonstrated that both ConvNeXt V2 and the LSKA module exhibited a superior detection accuracy and performance. The results are shown in Tables 2 and 3.

Table 2. Comparative experiment of different backbone network models.

Experimental Programs	mAP@0.5/%	Parameters/10^6	GFLOPs
YOLOv8n+AKConv	75.4	3.65	8.2
YOLOv8n+SAConv	77.2	3.51	7.2
YOLOv8n+fasterNet	75.1	2.7	5.9
YOLOv8n+C2f_DCN	76.3	3.07	7.9
YOLOv8n+ConvNext V2	78.8	3.6	7.8

Table 3. Comparative experiment of different attention mechanisms.

Experimental Programs	mAP@0.5/%	Parameters/10^6	GFLOPs
YOLOv8n+SE	82.9	3.01	8.2
YOLOv8n+GAM	83.8	4.64	9.5
YOLOv8n+CBAM	83.3	3.02	8.7
YOLOv8n+Biformer	82.2	3.27	18.6
YOLOv8n+LSKA	83.4	3.26	8.3

The data indicate that the YOLOv8n+ConvNeXt V2 configuration achieved an mAP@0.5 of 78.8%, which was higher than other models like AKConv, SAConv, fasterNet, and C2f_DCN.

While it had slightly higher parameters and GFLOPs, this enhanced accuracy suggests that ConvNeXt V2's advanced architecture, which likely incorporates larger convolutional kernels and deeper layers, effectively improved feature extraction capabilities without compromising speed too much.

The YOLOv8n+LSKA configuration achieved an mAP@0.5 of 83.4%, outperforming models using the SE, GAM, CBAM, and Biformer attention mechanisms. This indicates that the LSKA mechanism significantly enhanced the model's ability to capture and utilize contextual information across larger regions, leading to a better detection accuracy, particularly in complex environments. The balance between model complexity (parameters and GFLOPs) and performance also highlights the efficiency of the LSKA mechanism in improving feature extraction and fusion.

These results underscore the effectiveness of integrating ConvNeXtV2 and LSKA into the YOLOv8n framework, offering a more robust solution for tasks requiring a high detection accuracy, such as crack detection in challenging environments.

4.5. Ablation Study

An ablation study was conducted to evaluate the contributions of the improved modules to the overall performance of the model. To demonstrate the effectiveness of the two proposed improvements in the detection algorithm for cracks in fair-faced walls, YOLOv8n was taken as the base model. Precision, Recall, mAP@0.5, and confidence F1 were employed as evaluation metrics. Successively, the corresponding improvement measures were added to conduct the ablation experiments. The results are summarized in Table 4 below.

Table 4. Ablation experiment results.

Model	P/%	R/%	mAP@0.5/%	F1-Score/%	Inference Speed/ms
YOLOv8n	84.4	80.9	79.8	82.6	0.5
YOLOv8n+Convnextv2	87.2	83.2	78.8	85.2	2.8
YOLOv8n+LSKA	83.4	83.3	81	83.3	1.3
CL-YOLOv8	85.3	83.2	83.7	84.2	1.8

According to Table 2, integrating the ConvNeXt V2 model as the feature extractor for the YOLOv8n base model enhanced the precision of target detection by providing richer and more discriminative feature extraction capabilities. Although there was a slight decrease of 0.1% in the mAP value, it is evident from the table that both the precision and recall rates significantly improved, with a 2.8% increase in precision and a 2.3% increase in recall rate. This highlights that the adoption of ConvNeXt V2, with its larger convolutional kernels, deeper network structure, and more advanced design, enhanced the feature extraction and representation capabilities of the YOLOv8n model, enabling it to achieve more effective multiscale feature representation.

Additionally, the introduction of the Large Separable Kernel Attention (LSKA) mechanism into the SPPF feature fusion module of the YOLOv8n model resulted in a 1.2% increase in the mAP value, a 2.4% increase in the recall rate, and a 1% increase in the confidence F1 score. This indicates that the incorporation of the LSKA module enhanced the contextual information in features, aiding in the detection of challenging targets such as occlusions and truncations, thereby further improving the detection accuracy in various environmental scenarios.

By simultaneously introducing both improvement points into the YOLOv8n base model, comprehensive improvements in precision, recall, and mAP values were observed. The precision increased by 0.9%, the recall increased by 2.3%, and the mAP value increased by 3.9%, providing full evidence for the effectiveness of these algorithmic improvements.

5. Discussion

5.1. Result Verification

A visual comparison of the results of crack detection on a brick wall using YOLOv5s, YOLOv7-tiny, YOLOv8n, and the improved CL-YOLOv8 algorithm (from top to bottom) is presented in Figure 9. The CL-YOLOv8 algorithm showed a clear improvement in detection accuracy compared to the other models. This enhancement can be attributed to the integration of the ConvNeXt V2 model, which significantly improved feature extraction through deeper and more complex convolutional layers, and the LSKA mechanism, which enhanced the receptive field and feature correlations. These architectural improvements allowed CL-YOLOv8 to better capture subtle crack features, even in challenging lighting and environmental conditions, as demonstrated by the higher confidence scores and more precise crack localization.

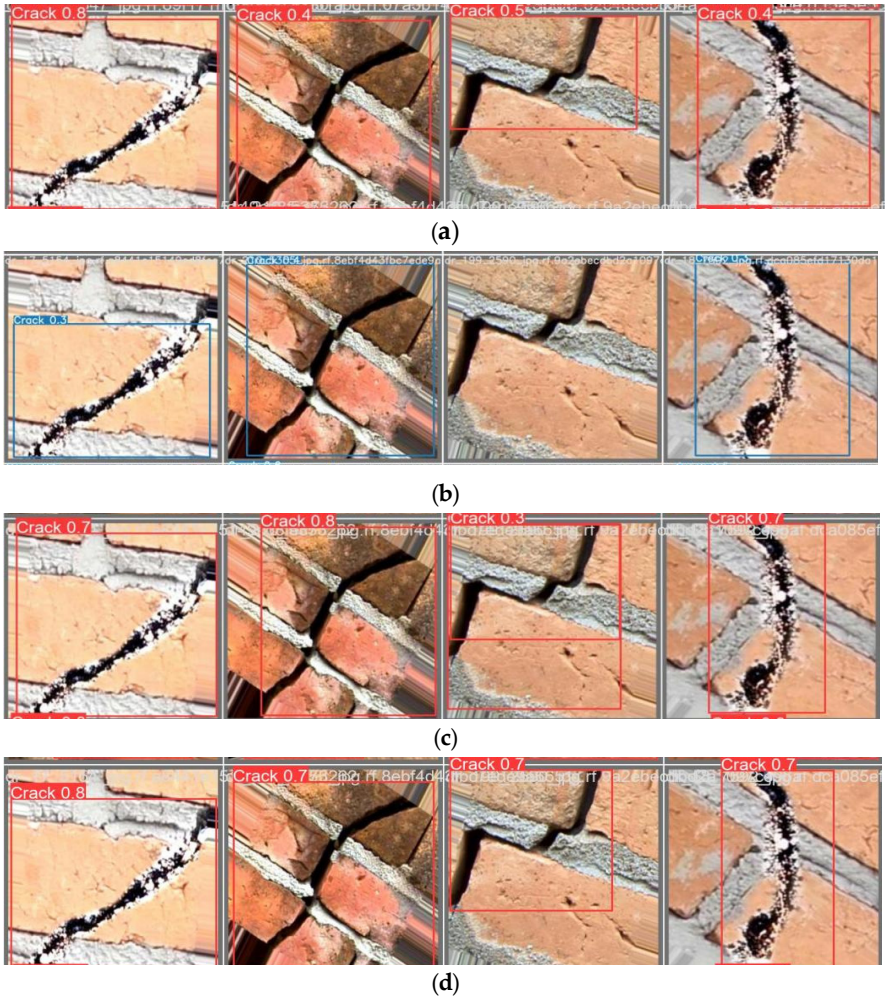

Figure 9. The detection effect of different models. YOLOv5s (**a**), YOLOv7-tiny (**b**), YOLOv8n (**c**), and CL-YOLOv8 (**d**).

From a numerical standpoint, CL-YOLOv8 achieved an accuracy of 85.3%, a recall rate of 83.2%, and a mean average precision (mAP) of 83.7%. These improvements in performance metrics were consistent with the visual results, as higher accuracy and recall rates translated into more reliable crack detections. In contrast, YOLOv5s and YOLOv7-tiny, while able to detect cracks, exhibited lower confidence scores and a reduced precision. This suggests that these models are less capable of distinguishing cracks from the surrounding texture or noise, particularly under less favorable conditions.

In comparison to existing methods, such as traditional image processing or non-AI-based techniques like ultrasonic or infrared thermography, the proposed algorithm demonstrated a superior adaptability to varying conditions. While these traditional methods are effective under controlled circumstances, their sensitivity to environmental changes is a limitation. By leveraging the strengths of advanced deep learning architectures, CL-YOLOv8 offers a more comprehensive solution for crack detection in diverse environments, which is crucial for maintaining the structural integrity of historical buildings. The test results for whether the fair-faced wall had cracks are shown in Figure 9.

5.2. Limitations and Challenges

This study successfully enhanced crack detection accuracy across different scenarios by introducing the LSKA attention mechanism and the ConvNeXtV2 model. This improvement primarily stemmed from enlarging the feature receptive field and enhancing inter-feature correlations to boost the model performance. However, it is worth noting that, while improving accuracy, it also led to an increase in parameters and computational complexity, posing limitations to achieving real-time detection. To realize real-time detection on device platforms, deploying lightweight models with faster inference speeds, achieved through techniques like pruning and quantization, becomes essential. Therefore, this will also be the focus of future research in this study.

6. Conclusions

This paper presents an improved YOLOv8 model tailored to the detection of cracks in brick walls. Several key contributions and findings emerged from this research, as follows:

1. Backbone Network Optimization: By replacing the original YOLOv8 backbone network with the more efficient ConvNeXt V2 model, the feature extraction capability was significantly enhanced. This modification improved the model's ability to detect subtle crack features, leading to a better performance in diverse environmental conditions.
2. Enhanced Attention Mechanism: The introduction of the Large Separable Kernel Attention (LSKA) mechanism into the SPPF structure further strengthened feature correlation and expanded the receptive field. This resulted in more accurate crack localization and an improved detection efficiency.
3. Superior Detection Performance: The proposed CL-YOLOv8 model demonstrated a superior performance compared to existing models, such as YOLOv5s, YOLOv7-tiny, and other YOLOv8 variants. It achieved higher accuracy, recall, and mAP values, which are critical for real-time crack detection on fair-faced brick walls. This capability will provide technical support for the early detection and timely repair of wall cracks, reducing potential safety risks and extending the lifespan of buildings.
4. Future Work on Crack Segmentation: While the current focus was on crack detection, future work will explore the task of crack segmentation. By segmenting detected cracks and extracting their skeletons, we will quantify geometric properties such as crack length, width, and area. Integrating a morphological analysis will enable a more comprehensive evaluation of crack damage, providing a deeper understanding of structural risks.

In summary, the improvements made to YOLOv8 enhanced its feature extraction and detection capabilities, offering a more robust solution for wall crack detection. These contributions lay a strong foundation for future research on crack segmentation and damage assessment in building structures.

Author Contributions: Conceptualization, Q.L.; Methodology, Q.L.; Software, G.Z.; Validation, G.Z.; Investigation, G.Z.; Resources, P.Y.; Data curation, P.Y.; Writing—original draft, G.Z.; Writing—review & editing, P.Y.; Supervision, P.Y.; Project administration, Q.L.; Funding acquisition, Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (61871260).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request. The data are not publicly available due to privacy.

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

References

1. Wojciechowska, G.; Bednarz, Ł.J.; Dolińska, N.; Opałka, P.; Krupa, M.; Imnadze, N. Intelligent Monitoring System for Integrated Management of Historical Buildings. *Buildings* **2024**, *14*, 2108. [CrossRef]
2. Golewski, G.L. Enhancement fracture behavior of sustainable cementitious composites using synergy between fly ash (FA) and nanosilica (NS) in the assessment based on digital image processing procedure. *Theor. Appl. Fract. Mech.* **2024**, *131*, 104442. [CrossRef]
3. Hamishebahar, Y.; Guan, H.; So, S.; Jo, J. A Comprehensive Review of Deep Learning-Based Crack Detection Approaches. *Appl. Sci.* **2022**, *12*, 1374. [CrossRef]
4. Marin, B.; Brown, K.; Erden, M.S. Automated Masonry crack detection with Faster R-CNN. In Proceedings of the IEEE 17th International Conference on Automation Science and Engineering (CASE), Lyon, France, 23–27 August 2021; pp. 333–340. [CrossRef]
5. Chen, W.; Chen, C.; Liu, M.; Zhou, X.; Tan, H.; Zhang, M. Wall Cracks Detection in Aerial Images Using Improved Mask R-CNN. *Comput. Mater. Contin.* **2022**, *73*, 767–782. [CrossRef]
6. Pratibha, K.; Mishra, M.; Ramana, G.V.; Lourenço, P.B. Deep Learning-Based YOLO Network Model for Detecting Surface Cracks During Structural Health Monitoring. *Struct. Anal. Hist. Constr.* **2023**, *47*, 179–187. [CrossRef]
7. Su, P.; Han, H.; Liu, M.; Yang, T.; Liu, S. MOD-YOLO: Rethinking the YOLO architecture at the level of feature information and applying it to crack detection. *Expert Syst. Appl.* **2024**, *237*, 121346. [CrossRef]
8. Nabizadeh, E.; Parghi, A. Deep Learning-Based Concrete Crack Detection Using YOLO Architecture. In *Artificial Intelligence and Smart Vehicles*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 182–193. [CrossRef]
9. Yang, B.; Wang, X.; Xing, Y.; Cheng, C.; Jiang, W.; Feng, Q. Modality Fusion Vision Transformer for Hyperspectral and LiDAR Data Collaborative Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2024**, *17*, 17052–17065. [CrossRef]
10. Zheng, Y.; Jiang, W. Evaluation of Vision Transformers for Traffic Sign Classification. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 3041117. [CrossRef]
11. Chen, C.F.R.; Fan, Q.; Panda, R. CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021. [CrossRef]
12. Ercolino, S.; Devoto, A.; Monorchio, L.; Santini, M.; Mazzaro, S.; Scardapane, S. On the robustness of vision transformers for in-flight monocular depth estimation. *Ind. Artif. Intell.* **2023**, *1*, 1. [CrossRef]
13. Xu, D.; Wu, Y.F.E. YOLO: A Feature Enhancement Network for Remote Sensing Target Detection. *Remote Sens.* **2021**, *13*, 1311. [CrossRef]
14. Chen, L.; Shi, W.; Deng, D. Improved YOLOv3 Based on Attention Mechanism for Fast and Accurate Ship Detection in Optical Remote Sensing Images. *Remote Sens.* **2021**, *13*, 660. [CrossRef]
15. Ran, Q.; Wang, Q.; Zhao, B.; Wu, Y.; Pu, S.; Li, Z. Lightweight Oriented Object Detection Using Multiscale Context and Enhanced Channel Attention in Remote Sensing Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 5786–5795. [CrossRef]
16. Zhou, J.; Zhang, B.; Yuan, X.; Lian, C.; Ji, L.; Zhang, Q.; Yue, J. YOLO-CIR: The network based on YOLO and ConvNeXt for infrared object detection. *Infrared Phys. Technol.* **2023**, *131*, 104703. [CrossRef]
17. Zhang, L.J.; Fang, J.J.; Liu, Y.X.; Feng Le, H.; Rao, Z.Q.; Zhao, J.X. CR-YOLOv8: Multiscale Object Detection in Traffic Sign Images. *IEEE Access* **2024**, *12*, 219–228. [CrossRef]
18. Shen, L.; Lang, B.; Song, Z. Infrared Object Detection Method Based on DBD-YOLOv8. *IEEE Access* **2023**, *11*, 145853–145868. [CrossRef]
19. Liu, S.; Wei, J.; Liu, G.; Zhou, B. Image classification model based on large kernel attention mechanism and relative position self-attention mechanism. *PeerJ Comput. Sci.* **2023**, *9*, e1344. [CrossRef]
20. Song, X.; Cao, S.; Zhang, J.; Hou, Z. Steel Surface Defect Detection Algorithm Based on YOLOv8. *Electronics* **2024**, *13*, 988. [CrossRef]
21. Hang, X.; Zhu, X.; Gao, X.; Wang, Y.; Liu, L. Study on crack monitoring method of wind turbine blade based on AI model: Integration of classification, detection, segmentation and fault level evaluation. *Renew. Energy* **2024**, *224*, 120152. [CrossRef]
22. Sohaib, M.; Jamil, S.; Kim, J.M. An Ensemble Approach for Robust Automated Crack Detection and Segmentation in Concrete Structures. *Sensors* **2024**, *24*, 257. [CrossRef]
23. Lau, K.W.; Po, L.M.; Rehman, Y.A.U. Large Separable Kernel Attention: Rethinking the Large Kernel Attention Design in Cnn. *Expert Syst. Appl.* **2023**, *236*, 121352. [CrossRef]
24. Guo, M.H.; Lu, C.Z.; Liu, Z.N.; Cheng, M.M.; Hu, S.M. Visual attention network. *Comput. Vis. Media* **2023**, *9*, 733–752. [CrossRef]

25. Wang, P.; Chen, P.; Yuan, Y.; Liu, D.; Huang, Z.; Hou, X.; Cottrell, G. Understanding Convolution for Semantic Segmentation. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018. [CrossRef]
26. Luo, D.; Xue, Y.; Deng, X.; Yang, B.; Chen, H.; Mo, Z. Citrus Diseases and Pests Detection Model Based on Self-Attention YOLOV8. *IEEE Access* **2023**, *11*, 139872–139881. [CrossRef]
27. Xu, Y.; Li, J.; Zhang, L.; Liu, H.; Zhang, F. CNTCB-YOLOv7: An Effective Forest Fire Detection Model Based on ConvNeXtV2 and CBAM. *Fire* **2024**, *7*, 54. [CrossRef]
28. Woo, S.; Debnath, S.; Hu, R.; Chen, X.; Liu, Z.; Kweon, I.S.; Xie, S. ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 16133–16142. [CrossRef]
29. Che, S.; Li, Z.; Shi, Z.; Gao, M.; Tang, H. Research on an underwater image segmentation algorithm based on YOLOv8. *J. Phys. Conf. Ser.* **2023**, *2644*, 012013. [CrossRef]
30. Li, Y.; Chen, Z.; Zhang, F.; Wei, Z.; Huang, Y.; Chen, C.; Zheng, Y.; Wei, Q.; Sun, H.; Chen, F. Research on detection of potato varieties based on spectral imaging analytical algorithm. *Spectrochim. Acta Part A Mol. Biomol. Spectrosc.* **2024**, *311*, 123966. [CrossRef]
31. Bian, E.; Yin, M.; Fu, S.; Gao, Q.; Li, Y. Part Defect Detection Method Based on Channel-Aware Aggregation and Re-Parameterization Asymptotic Module. *Electronics* **2024**, *13*, 473. [CrossRef]
32. Katsigiannis, S.; Seyedzadeh, S.; Agapiou, A.; Ramzan, N. Deep learning for crack detection on masonry façades using limited data and transfer learning. *J. Build. Eng.* **2023**, *76*, 107105. [CrossRef]
33. Hu, Z.C.; Wang, Y.; Wu, J.P.; Xiong, W.L.; Li, B.L. Improved Lightweight Rebar Detection Network Based on YOLOv8s Algorithm. *Adv. Comput. Signals Syst.* **2023**, *7*, 107–117. [CrossRef]
34. Wang, X.; Gao, H.; Jia, Z.; Li, Z. BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors* **2023**, *23*, 8361. [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.