engineering proceedings

MDPI

Proceeding Paper

Masonry Structure Wall Crack Identification Based on Dual-Scale Convolutional Neural Networks †

Ziwei Ma [1], Haiyang Li [1], Xiaoting Wang [1,2,*] and Tao Wang [1]

[1] Institute of Engineering Mechanics, China Earthquake Administration, Harbin 150080, China; 18322339623@163.com (Z.M.); lhy@iem.ac.cn (H.L.); wangtao@iem.ac.cn (T.W.)
[2] Key Laboratory of Life Search and Rescue Technology for Earthquake and Geological Disasters, Ministry of Emergency Management of China, Beijing 100049, China
* Correspondence: raphael0323@126.com
† Presented at the 7th International Conference on Civil, Architecture and Disaster Prevention and Control, Dali, China, 30 January–1 February 2026.

Abstract

This paper proposes a convolutional neural network (CNN)-based image recognition method for crack detection in masonry walls to improve accuracy and efficiency. A sample database is constructed using laboratory-captured images of cracked masonry walls, augmented through rotation, mirroring, gamma correction, contrast adjustment, and brightness modification. A dual-scale CNN architecture is designed by integrating the characteristics of VGG-16 and GoogleNet, enabling multi-scale feature extraction for load-bearing masonry wall images. The network adopts two input scales: a modified VGG-16 branch with 100 × 100 inputs and an optimized GoogleNet branch with 50 × 50 inputs. Both branches undergo feature extraction and classifier training. Leveraging transfer learning, the model is pre-trained on the large-scale ImageNet dataset and fine-tuned using our self-built masonry wall database. Multi-scale features from dual branches are weighted and fused to generate final crack recognition results. Experimental results demonstrate a training accuracy of 94.33%, precision of 94.78%, and F1-score of 94.68%, outperforming traditional CNN models. The proposed method achieves high recognition accuracy for masonry load-bearing wall cracks and provides reliable technical support for practical engineering applications.

Keywords: masonry structures; computer vision; convolutional neural network; crack detection; multi-scale feature extraction

1. Introduction

As infrastructure and construction have skyrocketed in China, brick-and-mortar structures have become a staple in the nation's building boom, particularly in seismic hotspots where they are even more prevalent. Unfortunately, these masonry wonders can take a pounding during tremors. Take the 2008 Wenchuan quake [1], for instance: in the hardest-hit zones, the masonry edifices were riddled with cracks and suffering major damage, whereas in less intense zones, the harm was relatively minor. The same was true for the 2014 Ludian quake and the 2022 Luding quake [2,3], where masonry buildings took a beating across the board. The first telltale sign of a building's woe is the appearance of cracks. Gathering and assessing post-earthquake crack data and ensuring structural integrity is crucial, especially for swift emergency evaluations and repairs [4].

Check for updates

Academic Editor: Mohamed Abdelkader Ismail

Published: 2 September 2026

Copyright: © 2026 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.

In masonry structures, fissures represent the most prevalent form of deterioration and provide a critical foundation for evaluating structural integrity. The safety of buildings must be assessed in compliance with established regulations, taking into account both the structural configuration and the nature of these cracks. As noted by Huang Zhenghua [5] in 2008, fractures in masonry are instrumental in gauging a building's safety. When cracks emerge, it often signals that the structure has reached its maximum load-bearing capacity, necessitating thorough inspection and risk analysis to decide if reinforcement measures are warranted. Per the "Earthquake Field Work Part 3: Survey Specifications," post-seismic evaluations should concentrate on the most affected levels and the orientation of wall fractures. During rapid safety checks, attributes like crack direction and length help identify structural compromise, while crack width influences decisions about repairing load-bearing walls. Metrics such as crack width, length, depth, and orientation serve as vital benchmarks for classifying a building's hazard rating. These parameters, combined with national guidelines, enable a comprehensive assessment of crack severity and inform the overall damage analysis.

In the wake of an earthquake, seasoned experts often have to rely on manual tools to inspect the fissures in structures. This old-school technique is not precisely reliable, tends to have room for error, and is both labor-intensive and a real time sinker, not to mention the safety hazards involved. Generally, the safety checks post-disaster are wrapped up in about a 10-day span, a process that requires a mountain of effort and heaps of pressure on the inspectors. Take the Wenchuan earthquake for instance; the assessment zone in Guangyuan City sprawled over 444,000 square meters, meaning each worker had to evaluate about 14,800 square meters daily—a massive load. These conventional methods are more of a drag, especially when you are racing against the clock and trying to manage a colossal number of buildings to inspect. Thanks to the advancements in artificial intelligence, the latest deep learning, and computer vision tech have been unleashed in civil engineering, making for a lightning-fast and efficient identification of any crack-related damage.

CNNs, or Convolutional Neural Networks, are a popular breed of deep neural networks that have gained traction in the realms of image classification and object detection. In the pursuit of innovative methods, Lei Zhang and his team embraced deep CNN techniques for the detection of road cracks, harnessing the power of features extracted from affordable sources such as smartphone sensors [6]. It is interesting to note that much of the crack detection research tends to concentrate on structures like bridges and roads, leaving the masonry wall crack detection playing second fiddle. Enter Liu Shengxin, who took a different approach with semantic segmentation for the detection of masonry cracks, only to find that the limitations of the negative sample database limited the model's adaptability [7]. NAV RAJ BHATT came along with a solution that blended CNNs and Grad-CAM to classify and pinpoint masonry damage, yet there remains a gap in dealing with intricate details [8]. Zhang Hanyu, on the other hand, employed Mask-RCNN for the masonry crack detection [9], intertwining the specifics of building location with crack characteristics to evaluate structural integrity. Yet, despite these advancements, practical applications reveal that accuracy can vary due to the uniqueness of building structures and the variables in filming conditions. All told, CNNs have proven to be a valuable tool in masonry crack detection, significantly enhancing our ability to pinpoint and categorize cracks with greater precision, which in turn offers robust data essential for conducting structural safety evaluations.

In light of the critical need to evaluate the structural integrity of masonry walls through image analysis, this study tackles the challenge of identifying cracks in load-bearing masonry by introducing a novel approach using Convolutional Neural Networks (CNNs). The goal is to step up the game in terms of both precision and speed for crack detection. Leveraging a dataset of crack images from block masonry walls gathered in

lab-based experiments, we implemented a dual-scale CNN algorithm to pull out multi-scale features from the imagery. We crafted and fine-tuned two CNN models with varying input dimensions, drawing inspiration from VGG-16 and GoogleNet architectures through transfer learning techniques. The findings demonstrate that this strategy hits the mark with impressive accuracy and rapid processing times for crack identification, making it a solid bet for practical engineering use by offering dependable technical backing.

2. Masonry Damage Database

2.1. Data Source

To build a robust dataset for training and validating models, this investigation draws on cyclic loading experiments carried out earlier by our research group. We gathered and curated 147 images from tests on diverse masonry specimens, such as unreinforced masonry walls, walls incorporating structural columns, and those made from recycled concrete solid brick. In evaluating masonry load-bearing walls, particular attention was paid to details like crack dimensions, orientation, quantity, and spatial arrangement. The assessment of wall damage hinges mainly on crack patterns—whether vertical, horizontal, or diagonal. Damage severity is gauged by considering both the direction and extent of cracking, while decisions regarding repair and reinforcement strategies are tied closely to measurements of crack width.

Consequently, a comprehensive image repository documenting masonry component damage was compiled, featuring diverse backgrounds alongside a wide spectrum of crack characteristics in terms of morphology and dimensions. This investigation systematically gathered masonry test samples representing varied shear-span ratios, mortar strength grades, brick classifications, void percentages, and other prevalent parameters. Table 1 enumerates the specimen categories, corresponding loading design specifications, and quantitative image data. The masonry test series chosen for this research were constructed at full scale and underwent cyclic static loading tests in compliance with established standard protocols. As a result, the damage patterns and failure mechanisms captured in the database authentically represent the visual signatures of masonry elements under genuine seismic conditions, thereby guaranteeing the database's relevance and practical utility for real-world applications.

Table 1. Masonry damage image database specimen design loading information and sample quantity statistics.

Specimen Type	Presence of Holes	Shear Span Ratio	Mortar Strength (MPa)	Quantity
Common brick masonry wall	No	1.0	7	13
	No	0.6	7	19
	No	0.8	3.3	10
	yes	0.8	3.3	21
Horizontal wall with structural columns	No	0.8	3.3	14
Vertical wall with structural columns	Yes	0.8	3.3	13
Recycled concrete solid brick masonry wall	NO	0.6	7.4	28
	NO	1.0	7.4	29

2.2. Data Selection

To build a robust database capturing varied masonry textures, lighting scenarios, image clarity, perspectives, intricate backdrops, obstructions, and crack configurations, it

is worth noting that deep learning systems thrive on data—their performance hinges on both the richness and volume of input. In crafting this dataset, several strategies were employed to broaden sample variety: Initially, as outlined in Table 1, a broad assortment of standard masonry samples was gathered, spanning differences in shear-span ratios, mortar durability, brick varieties, porosity, and additional attributes. Next, as outlined in Figure 1, throughout the loading experiments, numerous cameras documented specimen deterioration from multiple vantage points, yielding images with a mix of illuminations, resolutions, and angles. Furthermore, to bolster crack detection resilience amid cluttered environments, photos were handpicked to include busy settings with elements like cables, measurement tools, and testing apparatus, adding layers of complexity and occlusion. This comprehensive database underpins the convolutional neural network's ability to generalize effectively. It was then split randomly into training, validation, and test subsets following a 4:1:1 ratio to ensure balanced evaluation.

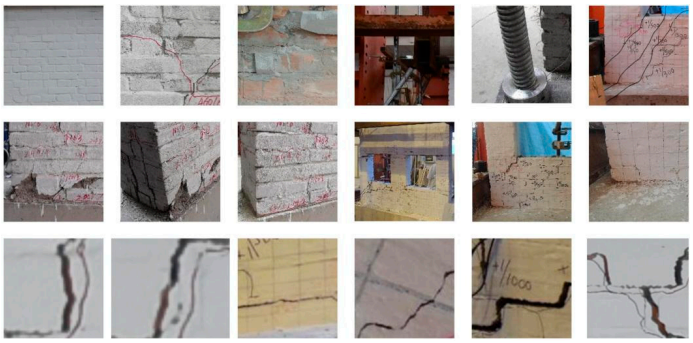


Figure 1. Composition of database samples.

2.3. Data Processing

To create two databases at different scales, the initial set of 147 images was partitioned into smaller 100 × 100 or 50 × 50 pixel subsets, each manually classified as containing cracks ("1") or not containing cracks ("0"), as illustrated in Figure 2. The masonry dataset incorporated a wide array of crack characteristics, spanning diverse lengths, widths, and angles. To beef up the database and enhance sample diversity, we applied various image processing techniques including rotation, mirroring, gamma correction, contrast adjustments, and brightness modifications. The core methodology for generating dataset samples from these database images involved randomly selecting sampling center points, rotation angles, scaling ratios (ranging from 0.9 to 1.1—keeping in mind that excessive scaling might compromise crucial crack features), and flipping axes to mimic variations in camera perspectives, thereby bolstering the dataset's diversity. Additionally, we employed techniques such as hue and saturation adjustments along with gamma correction to emulate lighting variations during photography, which in turn improved the overall representativeness of the dataset samples.

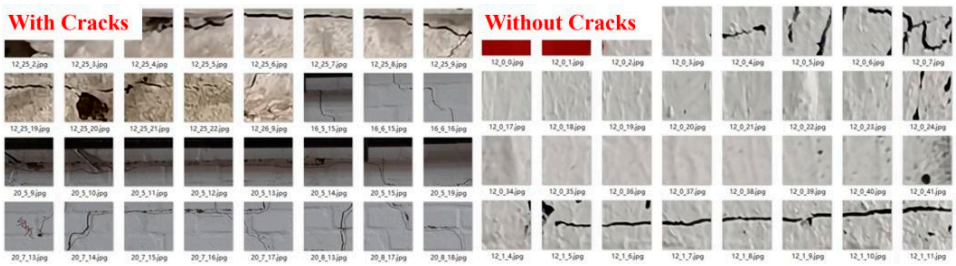


Figure 2. Label Classification of Masonry Wall Sample Database.

3. Crack Recognition Methods

3.1. General Process of Crack Recognition

The entire process of recognizing cracks through digital imaging involves several key steps (Figure 3): obtaining the image, prepping it up, identifying the cracks, tidying them with some morphological tricks, and finally, crunching the crack data. These cracks usually give you the red flags of altered brightness or color, unique grayscale traits, and they tend to stretch out in those long, continuous geometric patterns [10]. Because of the environment's fickle nature, these cracks in the pictures can cause difficulties with their poor contrast and lighting problems. During the prep stage, we get the image in tip-top shape with a good dose of enhancement and noise-killing techniques, which up the signal-to-noise game and bring the lighting up to the same level. The noise-busting morphological edge detection algorithm cooked up by Lei Wei and the crew has been a game-changer for dealing with the denoising and edge-saving conundrums in crack images [11].

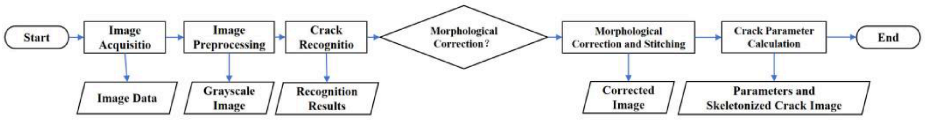


Figure 3. General process of crack identification.

The process of identifying cracks involves three key stages: preparation, detection, and refinement [12]. Initially, during preparation, the original image is segmented into manageable chunks that work well with the classification system. Song Beibei's approach, which leverages FCM and morphological techniques, proves particularly adept at uncovering subtle cracks with low contrast and fine details. Moving to the detection phase, the system pinpoints where cracks appear within these sub-images. Here, Xu Wei's saliency-based method offers enhanced precision in crack identification. As computer vision and artificial intelligence continue to advance, cutting-edge techniques like the deep learning framework combining YOLOv3 and MobileNet have entered the crack detection landscape. Finally, in the refinement stage, various crack characteristics—including width, length, and orientation—are carefully examined, with deep learning techniques employed to calculate relevant parameters. Given that geometric distortions can skew these measurements, geometric correction becomes essential. Liu's innovative 3D reconstruction-assisted correction method effectively removes perspective distortions [13–15], ensuring more reliable crack analysis [13–15].

3.2. Crack Detection

On one hand, the sample annotation of the CNN database is relatively simple, and on the other hand, CNN-based methods can utilize sliding window techniques and Digital Image Processing Techniques (DIPTs) to extract crack information from the background at the pixel level. Therefore, this paper adopts the classic CNN-based crack recognition method.

3.2.1. Introduction to CNN Model

CNN can be used for crack detection in three ways: image classification, object localization, or pixel-wise segmentation. The main goal of this paper is to train a CNN classifier to detect whether a sub-image contains cracks or not. There are various CNN architectures, such as AlexNet, GoogLeNet, ResNet18, VGG-16, etc. In this paper, a transfer learning approach is employed, modifying a few layers of the CNN configuration according to the dataset for training optimization. Generally, a CNN architecture includes the input layer, learning layer, and output layer. The input layer reads the image and passes it to the learning layer. The learning layer performs convolution operations by applying filters to

extract image features. The output layer uses the features extracted by the learning layer to classify the image into target categories. The new neural network can be trained by assigning target categories to images in the training dataset and iteratively modifying the filter values through backpropagation until the desired accuracy is achieved.

3.2.2. Dual-Scale Convolutional Neural Network

However, the width of cracks can vary widely, making it difficult for a single CNN to detect all cracks simultaneously. To address this issue, Ni, Zhang, and Chen proposed a dual-scale CNN to improve the detection performance for both fine and wide cracks. The dual-scale CNN consists of improved GoogleNet and ResNet, with image sizes of 224 × 224 pixels and 32 × 32 pixels as standard inputs, respectively. Considering the above conditions, this paper adopts a dual-scale convolutional neural network model for crack recognition. This approach can enhance the crack recognition speed while ensuring the recognition rate. Based on the differences in the two CNN scales, two independent databases are established, making it easier to train CNN models with different grid sizes. The model training and validation are conducted using the two databases of different sizes described in Section 2.3 of this paper. Afterward, a few original images not included in the database are used to test the performance of the classifier. Since the size of the original images is typically larger than 100 × 100 pixels, the sliding window technique is employed to scan the entire image, retaining the areas identified as cracks while discarding other background regions. This allows for obtaining the basic distribution of cracks.

3.2.3. Dual-Scale Convolutional Neural Network Model

Through empirical observation, the study found that using CNN alone (such as 100 × 100 pixels) for large-scale crack detection, based on Otsu threshold segmentation, tends to overlook finer cracks in the image. Therefore, by using GoogleNet with a smaller window, cracks can be located within a smaller field of view, enabling more accurate identification of fine cracks.

The first CNN is named MasonryCrackNet, which is a redesign of VGG16 with an input of 100 × 100 × 3 image classification model, divided into two parts: feature extraction and classifier. The feature extraction part uses the modified VGG16, which includes multiple convolutional layers and pooling layers to progressively extract the deep features of the image. The classifier part consists of multiple fully connected layers, ReLU activation functions, and Dropout layers, which are used to classify the extracted features. The second CNN refers to the classic GoogleNet network architecture with efficient classification capability. This network has an input designed as a 50 × 50 × 3 image classification model. Multi-scale features are extracted through GoogleNet's feature extraction layers and Inception modules, combined with adaptive average pooling and a two-stage classifier. The output category of the VGG16 and GoogleNet pre-trained models is 1000. Since masonry crack detection is essentially a binary classification task, the output layer of the prototype network architecture is adjusted to 2 classes (crack and non-crack).

The process of crack detection using the dual-scale convolutional neural network involves several steps (Figure 4). First, the input raw image undergoes preprocessing, where its size is adjusted to meet the input requirements of the model, and it is divided into multiple small blocks. Since a dual-scale model is used, the image is divided twice: once for input into the large-scale model (100 × 100) and once for input into the small-scale model (50 × 50). The large-scale (100 × 100) main model (MasonryCrackNet) performs a sliding window traversal to detect cracks in the image. Next, the small-scale (50 × 50) sub-model conducts a secondary scan of the image to obtain more detailed detection. Finally, the results are weighted and fused, and the final binarized mask image is returned. This process

effectively identifies crack areas in the image. By combining multi-scale information, it fully utilizes the global perspective of the large-scale model and the detail-capturing ability of the small-scale model, significantly improving the accuracy and robustness of crack detection.

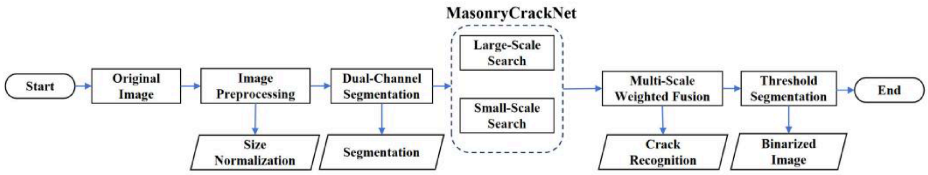


Figure 4. Flowchart of dual-scale CNN crack identification.

3.3. Post-Processing of Cracks

The image obtained after processing in Section 3.2, which contains the crack regions in binary form, undergoes further feature extraction based on Digital Image Processing Techniques (DIPT). Due to the complex nature of cracks in masonry load-bearing walls, this study selects the image filter method based on two-dimensional convolution operations for crack width measurement. This algorithm performs convolution between the crack and a series of strip templates at different angles. When the convolution result reaches a minimum, it is considered that the strip template is perpendicular to the crack at that location. The convolution result at this point is the product of the strip width of the convolution kernel and the crack width. The crack width can be obtained from the strip width of the convolution kernel. The crack length is estimated using a skeletonization method, which calculates the cumulative distance between adjacent points along the skeleton lines. The crack angle is determined using the average overall orientation of the skeleton points, calculated by averaging the angles between all adjacent points.

3.3.1. Width Calculation

The image filter method based on two-dimensional convolution operations performs convolution between the crack and a series of strip templates at different angles. When the convolution result reaches its minimum, it is considered that the strip template is perpendicular to the crack at that location. The convolution result at this point is the product of the strip width of the convolution kernel and the crack width. The crack width can be obtained from the strip width of the convolution kernel.

The strip template is set with a size of $W \times W$, and the strip consists of a series of fixed-length straight lines in the horizontal or vertical direction, each with a length of w_s. The angle between the strip and the horizontal direction is θ_s, which is uniformly distributed in the range of 0° to 180°. The angular increment used in this study is $\delta_s = 5°$. When the strip's angle is within the range of [0, 45] or [135, 175], w_s represents the vertical width of the strip; when the strip's angle is within the range of [50, 130], w_s represents the horizontal width of the strip. To obtain the width at any position of the crack, a pixel on the crack skeleton line is taken as the center point. A local region of the binary crack image with dimensions W × W is extracted and convolved with strip templates at various angles. The template convolution essentially results in the number of pixels in the overlapping region of the strip template and the crack, which can also be considered as the area of the overlapping region, denoted as $A(\theta_s)$, When the convolution result reaches its minimum value, it can be approximated that the strip corresponding to θ_s is perpendicular to the crack. The crack width w_s can be calculated using the following equation:

$$w_c = \frac{\min(A(\theta_s))}{w_s \cdot \cos\left(argmin_{\theta_s}(A(\theta_s))\right)} \quad (1)$$

Starting from either end of the crack skeleton line, the above operation is repeated along the crack skeleton line to obtain the width value at any position of the crack, as shown in Figure 5.

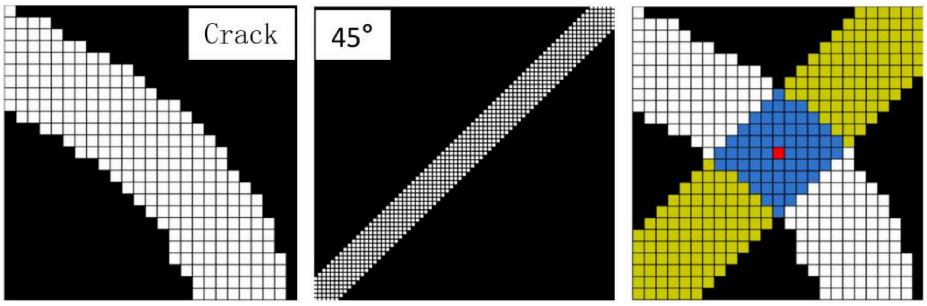


Figure 5. 45° Template convolution example.

3.3.2. Length Calculation

The crack length measurement in this paper is based on the morphological skeleton central axis transformation and the discrete geometric path integration principle. The skeletonization algorithm compresses the topological structure of the binary crack region into a single-pixel-wide central line, eliminating the interference of crack width in geometric representation, and establishes a skeleton model equivalent to the real crack shape. Then, along the skeleton line, the Manhattan-Euclidean hybrid distance (1 pixel for horizontal/vertical directions, $\sqrt{2}$ pixels for diagonal directions) between adjacent nodes is calculated pixel by pixel. The crack length is discretely approximated at the pixel level through path integration. This method is based on the raster topology of digital images, and its mathematical essence can be expressed as:

$$L = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (2)$$

In the formula, (x_i, y_i) represents the coordinates of the skeleton line nodes. The geometric features of the crack are preserved by minimizing the total length error of the skeleton line, achieving a conformal mapping of the crack geometry.

3.3.3. Angle Measurement

The crack angle analysis uses directional field statistical inference and global principal direction extraction methods. Based on the continuity feature of the skeleton line nodes, a local direction vector $\vec{v} = (\Delta x_i, \Delta y_i)$ is constructed using the coordinate differences between adjacent nodes. The direction angle of each vector is solved using the arctangent function:

$$\theta_i = arctan2(\Delta y_i, \Delta x_i) \times \frac{180}{\pi} \quad (3)$$

The global principal direction angle $\overline{\theta} = \frac{1}{n}\sum_{i=1}^{n} \theta_i$ is calculated through the arithmetic mean, and a probability distribution model of the crack orientation is established. This method combines differential geometry and statistical inference theory. Its core idea is to suppress local disturbance noise through low-pass filtering of the directional field, thereby extracting the macroscopic extension trend of the crack and meeting the characteristic representation needs of anisotropic structures.

https://doi.org/10.3390/engproc2026146022

4. Training and Testing

4.1. Model Hyperparameter Settings and Training Strategy

The masonry database is divided into two categories based on whether the image tiles contain cracks. When generating the model training set, 32 image tiles are read from the database for each class. The training batch size is set to 64, and the dataset is split into training and testing sets in a 7:3 ratio. The learning rate is set to 1×10^{-3}, the optimizer is set to stochastic gradient descent (SGD), with a momentum of 0.9, weight decay of 0.0001, and a learning rate scheduler that reduces the learning rate by 0.8 every 10 epochs. The loss function used is the Cross-Entropy Loss function. Each model is trained for 500 epochs to ensure recognition capability for both cracks and background. The models used in this study are all pre-trained on the large-scale ImageNet database. The pre-trained model, as a feature extractor, can extract basic features that represent natural objects (such as lines, edges, corner points, textures, color gradients, etc.) and more abstract deep features contained within these basic features. This study adopts the concept of transfer learning, where the parameters of the pre-trained model are imported into the model, and then fine-tuned with the specific images collected from various experiments to further learn from the information contained in this study's database based on the "knowledge" previously learned.

4.2. Model Architecture

This study first employed three convolutional neural network (CNN) models for training and comparison: AlexNet, GoogleNet, and the improved VGG16 model. AlexNet (2012) was the first convolutional neural network variant applied to image classification. It learns from the features themselves and introduces ReLU, Dropout, and data augmentation techniques, significantly improving the accuracy of image classification. GoogleNet is a novel deep learning architecture that greatly enhances computational efficiency and classification performance through the innovative Inception module, while reducing the number of parameters. VGG16 is a classic convolutional neural network model that improves feature extraction and classification performance by increasing the network depth.

In this study, due to the smaller input image size, different modifications were made to the CNN models used (Figure 6). The standard GoogleNet architecture was improved by using the initial convolutional layers, max pooling layers, and some Inception modules from GoogleNet, while removing some of the deeper layers. This approach reduces the complexity of the model, speeds up the training process, and reduces the demand for computational resources. Additionally, an Adaptive Average Pooling layer (AdaptiveAvgPool2d) was added to improve the model's generalization ability. A 1×1 convolutional layer was then used for feature reduction, followed by two fully connected layers for classification. This design allows for more effective integration and utilization of the extracted features, thus improving classification accuracy. The complexity of the model was reduced while maintaining good feature extraction capability. The standard VGG16 model was also modified and renamed as MasonryCrackNet. The feature extraction part is the same as the VGG16 architecture, using 13 convolutional layers and 5 pooling layers. The last pooling layer was modified to 3×3, which more effectively reduces the feature map size while retaining more spatial information. An additional fully connected layer was added to the classifier part to enhance the model's expressive capability. The standard AlexNet network was also modified. The feature extraction part remains the same as the standard AlexNet, with five convolutional layers and three max pooling layers. A custom classifier was introduced, adding a Dropout layer to prevent overfitting. In the classifier part, the output of the first fully connected layer was changed from 4096 to 2048, a new fully

https://doi.org/10.3390/engproc2026146022

connected layer with 1000 neurons was added, and the output categories were changed to 2.

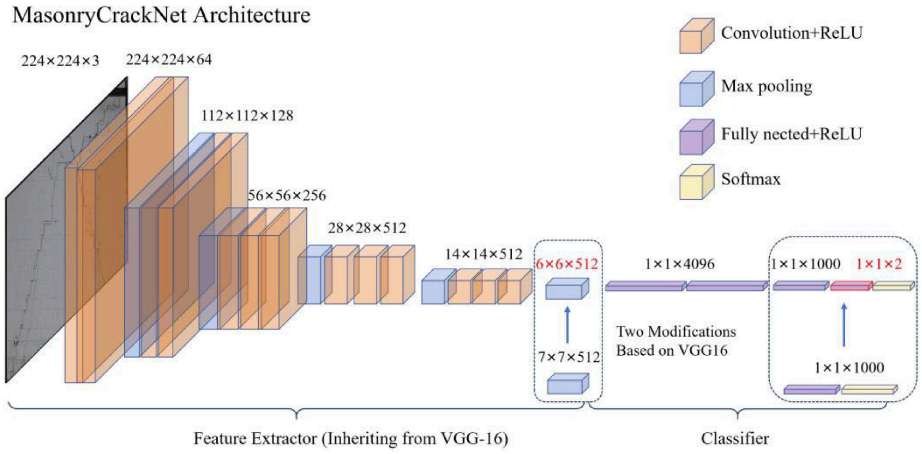


Figure 6. The architecture of VGG16 and MasonryCrackNet.

4.3. Model Evaluation

The improved models were trained, and after training, we performed model evaluation on the validation set to monitor the model's generalization performance. When evaluating the performance of convolutional neural network (CNN) models, we calculated the following metrics: Accuracy (Acc) is the degree to which the model's predicted results match the true labels; Specificity (True Negative Rate, TNR) is the proportion of correctly identified negative samples by the model; Precision (Pre) is the proportion of true positive samples among the results predicted as positive by the model; Recall (Re) is the proportion of correctly identified positive samples by the model; and the F1-score is the harmonic mean of precision and recall, used to evaluate the overall performance of the model. These metrics were used to assess the quality of the classification algorithm model.

As shown in the metrics in Table 2, GoogleNet performs the best across multiple metrics, particularly excelling in specificity (TNR), indicating its stronger ability in handling negative samples. The overall performance of Masonry_Crack_Net is similar to that of GoogleNet, while AlexNet performs noticeably worse than the other two models. Based on these results, this study selects Masonry_Crack_Net and GoogleNet as the foundation for the dual-scale model.

Table 2. Performance of three types of CNNs in identification.

Model \ Evaluation	ACC	TNR	Pre	Re	F1-Score
AlexNet	0.8047	0.8460	0.8097	0.8089	0.8058
GoogLeNet	0.9422	0.9578	0.9464	0.9467	0.9457
Masonry_Crack_Net	0.9433	0.9417	0.9478	0.9472	0.9468

4.4. Model Validation

Through experimental analysis, it was found that using CNN alone for large-scale crack detection often overlooks finer cracks in the image. When applying the model to the original large-sized crack images, it was observed that using a single CNN model struggles to cover all cracks in the image. As shown in Figure 7, using MasonryCrackNet is more suitable for detecting cracks with larger widths, while GoogleNet performs better in recognizing finer cracks. Based on this, a dual-scale convolutional neural network model is adopted for crack recognition, which can improve the crack detection rate. The original image is scanned twice: using MasonryCrackNet for sliding window traversal

with 100 × 100 large-scale image blocks and using GoogleNet for searching with 50 × 50 small-scale image blocks. By performing two scans on the original image with the dual-scale CNN model and combining the global perspective of the large-scale model with the detail-capturing ability of the small-scale model, the precision and robustness of crack detection can be significantly improved.

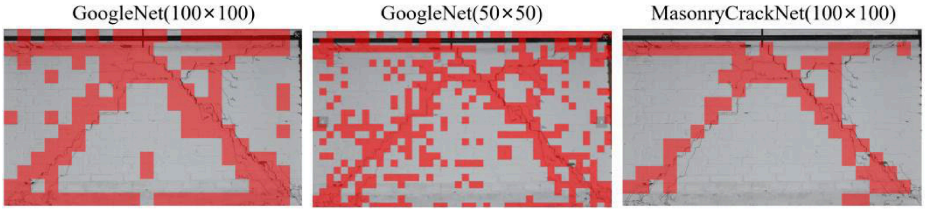


Figure 7. Model application.

4.5. Crack Feature Calculation

4.5.1. Crack Width Calculation

The image filter method based on two-dimensional convolution operations used in this study performs convolution between the crack and a series of strip templates at different angles. When the convolution result reaches its minimum, it is considered that the strip template is perpendicular to the crack at that location. The convolution result at this point is the product of the strip width of the convolution kernel and the crack width. The crack width can be obtained from the strip width of the convolution kernel, and the calculation results are shown in Table 3. The crack width measured by the comprehensive crack testing device is denoted as C_t, while the crack width obtained from the original crack image using the above algorithm is denoted as C_b.

Table 3. Crack width measurement results.

Shooting Angle		**30°**			**45°**	
Test points	1	2	3	1	2	3
$C_t(mm)$	0.19	0.23	0.37	0.19	0.23	0.37
$C_b(mm)$	0.212	0.216	0.427	0.253	0.263	0.405
$\lvert C_t - C_b \rvert / C_t$	12%	6%	15%	15%	33%	9%
Average deviation		11%			19%	

4.5.2. Crack Length Calculation

The crack length is determined by calculating the distance between adjacent pixels in the crack skeleton. Skeletonization is a process that reduces morphological structural elements to a width of just one pixel, maintaining the geometric properties of the shape. For the skeletonized crack image, the coordinates of all non-zero pixels are extracted. By traversing these coordinates, the distance between each pair of adjacent points is calculated. If the adjacent pixels are neighboring in the horizontal or vertical direction, the distance is 1; if they are adjacent in the diagonal direction, the distance is $\sqrt{2}$. Finally, these distance values are accumulated to obtain the total crack length.

4.5.3. Crack Angle Calculation

The calculation of the crack angle is based on analyzing the overall orientation of the skeleton line points. First, the coordinates of the skeleton line points are extracted. While traversing these coordinate points, the angle between each pair of adjacent points is calculated using the arctangent function. These angles are expressed in degrees, ranging from -180 degrees to 180 degrees. Then, the average of all these angles is computed to

obtain the overall directional characteristic of the crack. Based on this average angle, the crack type can be determined and classified as horizontal cracks (0–22.5 degrees and 157.5–180 degrees), vertical cracks (67.5–112.5 degrees), or diagonal cracks (other angle ranges).

5. Conclusions

This study proposes a masonry structure crack detection method based on a dual-scale convolutional neural network, addressing the issue that traditional single-scale CNNs struggle to simultaneously detect crack locations and widths in masonry load-bearing walls. Through three stages—preprocessing, crack detection, and post-processing—cracks are effectively identified and quantified. The use of a diversified database and image enhancement techniques has improved the model's generalization ability. Experimental results show that GoogleNet and MasonryCrackNet achieved recognition accuracies of 94.22% and 94.33%, respectively, on the test set. When tested with real large-sized images, the effectiveness of the method in practical applications was also validated. The image filter method based on two-dimensional convolution operations measures crack width by convolving strip templates at different angles. The crack length is estimated using the skeletonization method. The overall orientation average method of skeleton points is used to calculate the crack angle, providing important reference information for masonry structure damage assessment and repair. Future work can further optimize the model architecture and training methods to improve the accuracy and efficiency of crack detection.

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

Funding: This research was funded by Key Research and Development Program Special Project, grant number 2022YFC3003602; and the Scientific Research Fund of Institute of Engineering Mechanics, China Earthquake Administration, grant number 2021B03.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Only the data supporting the conclusions are presented in the article. Further raw data are available upon reasonable request from the corresponding author.

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

References

1. Li, Y.; Han, J.; Liu, L.; Zheng, N.; Wang, L.; Liu, J. "5·12" Wenchuan Earthquake Masonry Structure Housing Damage Survey and Analysis. *J. Xi'an Univ. Archit. Technol. (Nat. Sci. Ed.)* **2009**, *41*, 606–611. [CrossRef]
2. Pan, Y.; Peng, X.; Wang, T.; Shang, Q.; Wang, T. Seismic Damage Survey and Analysis of Medical Buildings in the 6.8 Magnitude Luding Earthquake. *J. Build. Struct.* **2024**, *45*, 14–29. [CrossRef]
3. Zhu, Z.; Chen, T.; Zou, Z.; Que, M. Seismic Damage Characteristics of Single-Bay Masonry Structures in the Lushan Earthquake. *J. Disaster Prev. Reduct. Eng.* **2021**, *41*, 203–210. [CrossRef]
4. Pan, Y.; Chen, J.; Bao, Y.; Peng, X.; Lin, X. Survey and Analysis of Seismic Damage to Village and Town Buildings in the 6.0 Magnitude Changning Earthquake. *J. Build. Struct.* **2020**, *41*, 297–306. [CrossRef]
5. Huang, Z.; Ge, Q. Identification and Treatment of Common Masonry Cracks in Buildings. *Shanxi Archit.* **2008**, *28*, 153–154. [CrossRef]
6. Zhang, L.; Yang, F.; Zhang, Y.D.; Zhu, Y.J. Road crack detection using deep convolutional neural network. In *Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP)*; IEEE: Piscataway, NJ, USA, 2016; pp. 3708–3712. [CrossRef]
7. Liu, S. *Research on Crack Recognition Technology of Block Masonry Walls Based on Fully Convolutional Neural Networks*; Harbin Institute of Technology: Harbin, China, 2020. [CrossRef]

8. Bhatt, N.R. *Post-Earthquake Damage Assessment of Masonry Walls Based on Convolutional Neural Networks*; Harbin Institute of Technology: Harbin, China, 2020. [CrossRef]
9. Zhang, H. *Research on Crack Recognition and Safety Evaluation of Masonry Structure Houses Based on Deep Learning*; Nanjing University of Science and Technology: Nanjing, China, 2022. [CrossRef]
10. Liu, Y.; Fan, J.; Nie, J.; Kong, S. Review and Prospect of Structural Surface Crack Identification Using Digital Image Methods. *J. Civ. Eng.* **2021**, *54*, 79–98.
11. Li, W.; Zhu, P. Research on Asphalt Pavement Crack Image Detection Algorithm. *Comput. Eng. Appl.* **2012**, *48*, 163–166+219.
12. Miao, P.; Srimahachota, T. Cost-effective system for detection and quantification of concrete surface cracks by combination of convolutional neural network and image processing techniques. *Constr. Build. Mater.* **2021**, *293*, 123549. [CrossRef]
13. Xie, Z. Research on Concrete Surface Crack Recognition Based on Fully Convolutional Networks. Master's Thesis, Northwest A&F University, Yangling, Shaanxi, China, 2022. [CrossRef]
14. Geng, H. *Crack Detection and Width Calculation Based on Computer Vision*; The Beijing University of Technology: Beijing, China, 2020. [CrossRef]
15. Lei, S.; Lin, J.; Huang, S.; Xiao, Q.; Haung, Y. Research on Concrete Surface Crack Target Recognition Based on Improved YOLOv3 Network. *Highway* **2024**, *69*, 270–275.

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.