Multidamage Identification in High-Resolution Concrete Bridge Component Imagery Based on Deep Learning

Hai-En Tang, S.M.ASCE[1]; Ting-Hua Yi, M.ASCE[2]; Song-Han Zhang, Ph.D.[3]; and Chong Li, Ph.D.[4]

Abstract: The identification of damage to concrete bridge surfaces is of great significance to maintaining the durability and reliability of bridges. However, it is difficult to identify small areas of damage, especially in high-resolution images. To conduct damage identification research in a targeted manner, this paper proposes a method of multidamage identification in high-resolution images based on You Only Look Once version 5 (YOLOv5). In this paper, a data set labeled with four types of damage (crack, spallation, hole, and rebar) is used for training, validation, and testing. To ensure that the network adequately learns the damage features, the high-resolution images are cropped into subimages via the autoadaptive window cropping method (AWCM) proposed in this paper. The cropping method can crop images according to the label information and protect the damage features from destruction during the cropping process. To avoid overfitting, it is necessary to balance the volume of different types of damage. After balancing, the count for each category is as follows: crack (4,980), spallation (5,225), hole (5,211) and rebar (5,020). The balanced data set can be used to train the deep learning network and construct the multidamage identification model. After identification, the subimages, and the prediction boxes of damages in them are restored to their original high-resolution images. The results show that the mean average precision (mAP) of all classes is 94.2%, and the values for cracks, spallation, holes, and rebars are 86.9%, 98.1%, 92.3%, and 99.4%, respectively, which indicates that the proposed method outperforms the other three methods (inputting original images directly, the sliding window cropping method, and the random centroid cropping method). **DOI: 10.1061/JPCFEV.CFENG-4671.**

Author keywords: Concrete bridge; Damage identification; Multidamage; Deep learning; High-resolution image.

Introduction

Bridges play an important role in modern transportation systems and road network frameworks. Concrete is one of the most used bridge materials (Zhang et al. 2022). During the construction and service of concrete bridges, various types of damage may appear on bridge surfaces, such as cracks, spallation, rebars, holes, and efflorescence. If damage is not detected in time, it may lead to a decrease in structural load-bearing capacity and even cause safety accidents. Therefore, timely and accurate identification of bridge surface damage is of critical importance to ensure the safety and durability of bridge structures (Du et al. 2023).

The common methods for identifying damage on bridge surfaces are artificial vision, data analysis, and image processing (Wan et al. 2023). Among these three methods, artificial vision is not only time-consuming and laborious but also requires inspectors with professional bridge knowledge. Moreover, methods based on data analysis rely on the deployment of sensors (Deng et al. 2020), which cannot detect arbitrary locations of bridges, and sensor faults also cause difficulties in monitoring. By comparison, image processing methods can identify damage more conveniently and quickly (Xu and Brownjohn 2018). To improve the robustness of damage identification, general global transform, and local edge detection methods, including the fast Haar transform (FHT), fast Fourier transform (FFT), and Sobel and Canny edge detectors (Abdel-Qader et al. 2003), are combined with damage contour identification. By combining the improved Sobel operator (Wang et al. 2019) and wavelet transform (Su and Gong 2020) to remove noise, the robustness of damage identification can also be improved. However, traditional image processing methods require artificially designed filters and edge detection operators to extract image features (Nishikawa et al. 2012; Wang et al. 2018), which makes it difficult to guarantee the stability of identification results (Yu et al. 2023). Methods based on machine learning can achieve higher identification precision and are used to identify specific structural damage, such as cracks (Wang et al. 2021; Mir et al. 2022), spallation (Cao et al. 2021; Hoang et al. 2021), holes (Zhao et al. 2021), and efflorescence (Fan and Chung 2022). However, when the image background is complex, the recognition effect of these approaches is limited.

Deep learning, as a subfield of machine learning, has been widely applied in recent years. Compared with traditional machine learning, deep learning can extract higher-level damage features, thus improving the accuracy of damage identification (Sun et al. 2020; Baduge et al. 2022). Researchers have modified different deep-learning networks for different data sets. A faster region-based convolutional neural network (Faster R-CNN)-based structural visual inspection method was proposed to identify five types of damage, namely, concrete cracks, two levels of steel corrosion, bolt corrosion, and steel plate delamination, and the results showed that the mean average precision (mAP) reached 87.8% (Cha et al. 2018). Another modified Faster R-CNN method was proposed for

[1]Master's Student, School of Civil Engineering, Dalian Univ. of Technology, Dalian 116023, China. Email: tanghaien1@126.com

[2]Professor, School of Civil Engineering, Dalian Univ. of Technology, Dalian 116023, China (corresponding author). Email: yth@dlut.edu.cn; yth@bucea.edu.cn

[3]Associate Professor, School of Civil Engineering, Dalian Univ. of Technology, Dalian 116023, China. Email: shz@dlut.edu.cn

[4]Senior Engineer, CCCC Highway Bridges National Engineering Research Centre Co., Ltd., No. 23 Huangsi St., Xicheng District, Beijing 100120, China. Email: lichong@bnerc.com

Note. This manuscript was submitted on September 12, 2023; approved on March 27, 2024; published online on June 28, 2024. Discussion period open until November 28, 2024; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Performance of Constructed Facilities***, © ASCE, ISSN 0887-3828.

the identification and localization of concrete cracks, concrete spallation, rebar exposure, and rebar bucking (Xu et al. 2019). As the number of bridge images to be identified increases, the speed of damage identification becomes increasingly important. A single shot multibox detector (SSD) network was used to identify linear cracks and corruptions (Maeda et al. 2018). Another method involving the introduction of a transfer learning method and batch regularization method to the You Only Look Once version 3 (YOLOv3) network was used to identify bridge cracks, spallation, honeycombs, and leaky reinforcements (Zhang et al. 2020). However, these methods, which improve the network structure, yield limited improvement in the mAP and are highly dependent on specific data sets, and thus lack universality.

By comparison, focusing on the processing of the data set can often greatly improve the identification accuracy of the network. Due to the limited computational power of computers, the resolution of the training set images for deep learning networks cannot be too high. However, images, whether captured by cameras or smartphones, have a high resolution that far exceeds the computational capacity of the network, so it is necessary to crop high-resolution images into many low-resolution subimages. Common image cropping methods include the sliding window cropping method (Chen et al. 2023) and the random centroid cropping method. The sliding window cropping method refers to setting a window of a specified size to slide-crop high-resolution images so that they can be cropped to subimages with an appropriate input image size for the network, avoiding the loss of damage features during the process of resizing. However, when the images are cropped by the sliding window, the damage feature information will also be cropped, resulting in inaccurate feature information learned by the network, which will increase the false identification rate and missing identification rate of the model. In contrast, the random centroid cropping method crops high-resolution images based on the location of the target boxes in the images. First, the center of each bounding box of damage labeled before is the center of the cropping window. Then, to avoid overfitting the center point position of the image, the center point of the cropping window is moved randomly. Finally, based on the center point of the cropping window, a subimage of a specified size is cropped. This method ensures the integrity of the damage during the cropping process. Due to the introduction of randomness, the subimages cropped out are different each time, resulting in different trained networks, which makes the identification effect unstable and uncontrollable.

Therefore, to solve the problem of identifying multiple types of damage in high-resolution images and ensure the universality and controllability of deep learning networks, a novel multidamage identification framework using an autoadaptive window cropping method (AWCM) is proposed in this paper. The paper is organized as follows. First, the theories and methods of the multidamage identification framework using the AWCM are introduced. Next, an annotated concrete bridge surface damage data set is constructed and balanced. This data set contains 1,012 images with a resolution of $3{,}000 \times 4{,}000$, including four types of damage: cracks, spallation, holes, and rebars. Then, the effectiveness of the proposed damage identification method is verified. Finally, the conclusions are presented.

Multidamage Identification Framework Using AWCM

In this section, a novel multidamage identification framework using the AWCM is proposed to meet the challenge of multidamage identification in high-resolution images. The framework includes three parts. First, the AWCM is used to crop high-resolution images into many subimages suitable for a deep learning network. Second, the YOLOv5 network (Wang et al. 2023; Zhao et al. 2022) is selected to train the subimages to construct a model that can identify multiple types of surface damage. Third, the identified subimages and their prediction boxes are restored to their original high-resolution images.

Autoadaptive Window Cropping Method for Multidamage Images

The sizes of the images collected by researchers are usually much larger than the input size required by the network, so it is necessary to crop high-resolution images into many subimages of suitable sizes. Otherwise, if high-resolution images are input directly into the network, they will be resized to fit the network. During the process of resizing, multiple pixels of the image will be integrated into one pixel, and much information will be lost. Therefore, the network cannot fully learn the feature information, resulting in a poor identification effect (Arulalan and Kumar 2023). Compared to traditional image cropping methods (including the sliding window cropping method and the random centroid cropping method), the AWCM proposed in this paper can keep the damage features intact. The size and location of the cropping window in the AWCM will auto-adaptively adjust according to the labeled bounding box of damage in the high-resolution images and the input size required by the network.

Size of Cropping Window

To protect the integrity of damage features, the cropping window must contain the bounding box of the damage. Therefore, the size of the cropping window should be the maximum of the bounding box size of the damage and the size required by the network. The size of the cropping window can be determined from Eqs. (1) and (2):

$$ww = \max(nw, bw) \tag{1}$$

$$wh = \max(nh, bh) \tag{2}$$

where ww = width of the cropping window; nw = width of the input image required by the network; bw = width of the bounding box of the damage; wh = height of the cropping window; nh = height of the input image required by the network; and bh = height of the bounding box of the damage.

Location of Cropping Window

To crop the images into subimages of appropriate sizes, the location of the cropping window should be adjusted according to the location of the bounding box and the size of the image. The cropping window should not exceed the scope of the image. The location of the cropping window can be determined from Eqs. (3) to (6), and an example is shown in Fig. 1.

When $x_{\min} + ww < W$, and $y_{\min} + wh < H$

$$\begin{aligned} x_0 &= x_{\min} \\ y_0 &= y_{\min} \\ x_1 &= x_{\min} + ww \\ y_1 &= y_{\min} + wh \end{aligned} \tag{3}$$

When $x_{\min} + ww \geq W$, $x_{\max} - ww \geq 0$, and $y_{\min} + wh < H$

$$\begin{aligned} x_0 &= x_{\max} - ww \\ y_0 &= y_{\min} \\ x_1 &= x_{\max} \\ y_1 &= y_{\min} + wh \end{aligned} \tag{4}$$

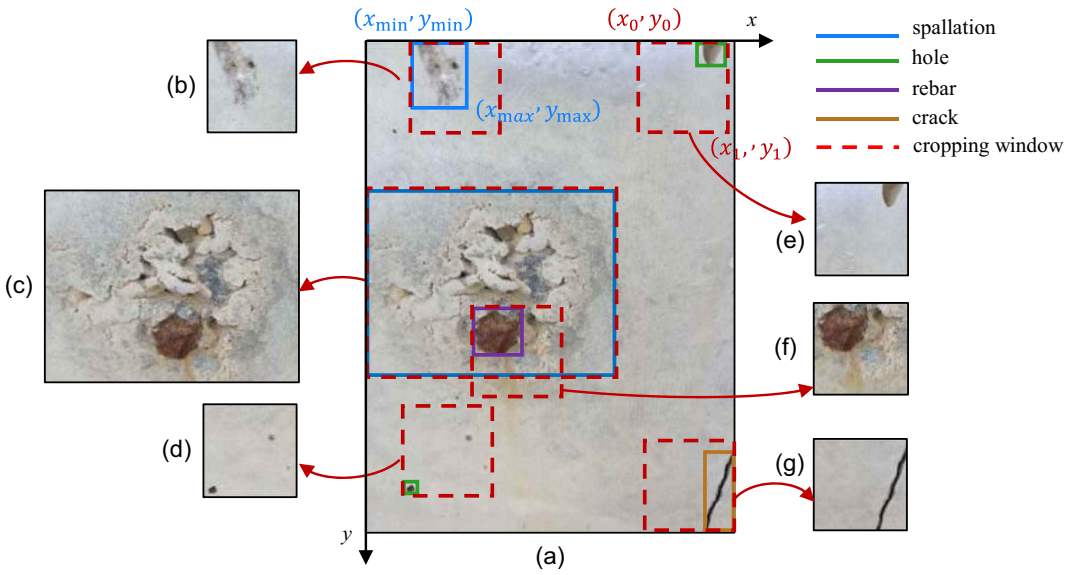

Fig. 1. Example of how to determine the location of the cropping window in the autoadaptive window cropping method: (a) the original image with 3,000 × 4,000 pixels; (b) a subimage with spallation; (c) a subimage with spallation and a rebar; (d) a subimage with a hole; (e) a subimage with a hole; (f) a subimage with a rebar; and (g) a subimage with a crack.

When $y_{min} + wh \geq H$, $x_{min} + ww < W$, and $y_{max} - wh \geq 0$

$$\begin{aligned} x_0 &= x_{min} \\ y_0 &= y_{max} - wh \\ x_1 &= x_{min} + ww \\ y_1 &= y_{max} \end{aligned} \tag{5}$$

When $x_{min} + ww \geq W$, and $y_{min} + wh \geq H$

$$\begin{aligned} x_0 &= x_{max} - ww \\ y_0 &= y_{max} - wh \\ x_1 &= x_{max} \\ y_1 &= y_{max} \end{aligned} \tag{6}$$

where x_{min} = minimum value on the x axis of the bounding box in the original image; x_{max} = maximum value on the x axis of the bounding box in the original image; y_{min} = minimum value on the y axis of the bounding box in the original image; y_{max} = maximum value on the y axis of the bounding box in the original image; x_0 = minimum value on the x axis of the cropping window; x_1 = maximum value on the x axis of the cropping window; y_0 = minimum value on the y axis of the cropping window; y_1 = maximum value on the y axis of the cropping window; W = width of the original image; and H = height of the original image.

Fig. 1 shows an example of the autoadaptive window cropping method. The locations of the cropping windows in Figs. 1(b, c, and f) are determined according to Eq. (3); the location of the cropping window in Fig. 1(e) is determined according to Eq. (4); the location of the cropping window in Fig. 1(d) is determined according to Eq. (5); and the location of the cropping window in Fig. 1(g) is determined according to Eq. (6).

Labeling Information Cropping Method

When the images are cropped, the labeling information is cropped simultaneously. To input the subimages with the correct label information into the network for training, it is necessary to convert the labeling information in the original high-resolution images into subimages. The new coordinate of the bounding box containing damage in the subimages can be determined from Eq. (7)

$$\begin{aligned} x'_{min} &= x_{min} - x_0 \\ y'_{min} &= y_{min} - y_0 \\ x'_{max} &= x_{max} - x_0 \\ y'_{max} &= y_{max} - y_0 \end{aligned} \tag{7}$$

where x'_{min} = minimum value on the x axis of the bounding box in the subimage; x'_{max} = maximum value on the x axis of the bounding box in the subimage; y'_{min} = minimum value on the y axis of the bounding box in the subimage; and y'_{max} = maximum value on the y axis of the bounding box in the subimage.

The proposed AWCM crops images according to the label information of the damages one by one. Although the whole label information of the corresponding damage is preserved, when there is more than one damage in the cropping window, the other label information should be processed appropriately. Therefore, a threshold is set to determine whether the label information of other damage in the cropping window should be preserved in subimages. The discriminant is given in Eqs. (8) and (9). and the new coordinate of the bounding box in the subimages can be determined from Eq. (10):

$$\text{IoB} = \frac{A \cap B}{B} \tag{8}$$

$$\text{IoB}_{threshold} \leq \text{IoB} \tag{9}$$

$$\begin{aligned} x'_{mino} &= \max(x_{mino} - x_0, 0) \\ y'_{mino} &= \max(y_{mino} - y_0, 0) \\ x'_{maxo} &= \min(x_{maxo} - x_0, ww) \\ y'_{maxo} &= \min(y_{maxo} - y_0, wh) \end{aligned} \tag{10}$$

where IoB = intersection of A and B over B; A = area of the cropping window; B = area of the damage's bounding box; $\text{IoB}_{threshold}$ = threshold of the intersection of A and B over B, in this paper, the $\text{IoB}_{threshold}$ is set to 30%. When $\text{IoB}_{threshold} \leq \text{IoB}$, the labeling

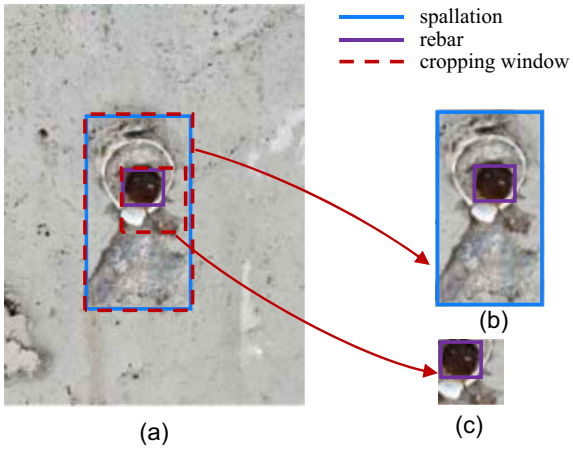

Fig. 2. Example of whether to preserve the label information of damages in their subimages: (a) the original image with 3,000 × 4,000 pixels; (b) a subimage with the label information of spallation and rebar; and (c) a subimage with the label information of rebar.

information of the damage should be reserved in subimages; inversely, when $\mathrm{IoB}_{\mathrm{threshold}} > \mathrm{IoB}$, the labeling information of the damage should be abandoned in subimages. $x'_{\mathrm{min}o}$ = minimum value on the x axis of the bounding box of other damage in the subimage; $x'_{\mathrm{max}o}$ = maximum value on the x axis of the bounding box of other damage in the subimage; $y'_{\mathrm{min}o}$ = minimum value on the y axis of the bounding box of other damage in the subimage; $y'_{\mathrm{max}o}$ = maximum value on the y axis of the bounding box of other damage in the subimage; $x_{\mathrm{min}o}$ = minimum value on the x axis of the bounding box of other damage in the original image; $x_{\mathrm{max}o}$ = maximum value on the x axis of the bounding box of other damage in the original image; $y_{\mathrm{min}o}$ = minimum value on the y axis of the bounding box of other damage in the original image; $y_{\mathrm{max}o}$ = maximum value on the y axis of the bounding box of other damage in the original image.

An example is given in Fig. 2. For Fig. 2(b), the IoB between the cropping window and spallation is 100%, while that of the rebar is also 100%, so the labeling information of the spallation and the rebar is reserved in Fig. 2(b); in Fig. 2(c), the IoB between the cropping window and spallation is less than 30%, while that of the rebar is 100%, so the labeling information of the spallation is abandoned and the labeling information of the rebar is reserved in Fig. 2(c).

YOLOv5 Model for Identifying Multiple Types of Damage

To determine the actual identification effect, a deep learning model need to be the medium. You Only Look Once version 5 (YOLOv5), which is currently used in many fields and has achieved good performance (Mukhiddinov et al. 2022; Wu et al. 2022; Yu et al. 2022), is selected for training and identifying of multiple types of damage.

Structure of YOLOv5

Based on differences in network depth and width, YOLOv5 can be divided into four network model versions: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Considering the model size, this paper gives priority to the fastest YOLOv5s.

After the images are input to the YOLOv5 network, the following processing will be carried out: mosaic data enhancement, image size processing, and adaptive anchor box calculation. Mosaic data enhancement uses a combination of 4 photos to enrich data diversity, and image sizing processing can resize the images to the input size required by the model. In addition, adaptive anchor box calculation is helpful to improve the detection speed.

The YOLOv5 network contains three parts: the backbone, neck, and head. The backbone of the YOLOv5 network is responsible for feature extraction from the input images, while the neck of the network is responsible for fusing the multiscale features of the feature map and transmitting these features to the head layer. Both the backbone and neck use CSPDarknet-53 to solve the problem of repeating gradient. By comparison, the head of the network is used to make the final regression prediction (Qiu et al. 2022). The structure of the YOLOv5 model for multidamage identification is shown in Fig. 3.

The YOLOv5 network uses GIoU as the network loss function, as shown in Eqs. (11) and (12):

$$\mathrm{IoU} = \frac{X \cap Y}{X \cup Y} \tag{11}$$

$$\mathrm{GIoU} = \mathrm{IoU} - \frac{|Z| - |X \cup Y|}{|Z|} \tag{12}$$

where IoU = intersection of X and Y over the union of X and Y; X = predicted bounding box and Y = ground-truth bounding box; Z = smallest box containing X and Y. GIoU not only pays attention to the overlapping area but also to other nonoverlapping areas, which can better reflect the degree of overlap between the two.

Model Training

A desktop computer is used as the processing platform, the operating system is Windows10 and the Pytorch framework and the YOLOv5 environment are built in the Anaconda3 environment. The program is written in Python 3.9, and the CUDA version is 10.1. For hardware, the processor is a 12th Gen Intel(R) Core (TM) i5-12400, the main frequency is 2.5 GHz, the memory is 16G and the graphics card is a NVIDIA GeForce RTX 2060.

For the training phase, the training and verification sets are input into the network, with a batch size of 16 and 300 epochs, respectively. Transfer learning is used to save time and improve the initial

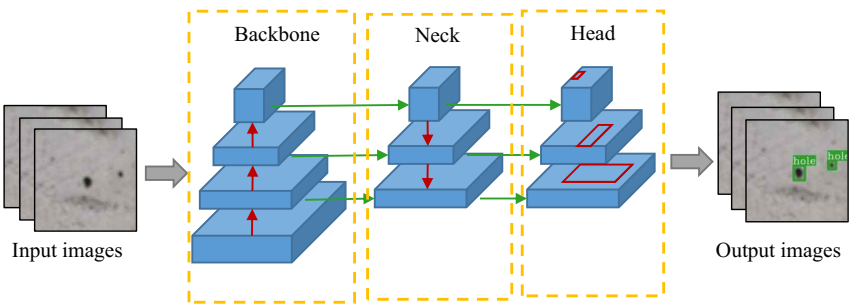

Fig. 3. Structure of the YOLOv5 model for multidamage identification.

J. Perform. Constr. Facil., 2024, 38(5): 04024026

effectiveness of the network. The network in this paper uses a model weight trained using the COCO data set, which includes more than 330,000 images, as the initial model weight for the identification of multiple types of damage. The input image resolution is 640 × 640 and the initial learning rate is 0.01. As for the momentum, it is set to 0.937 while the warmup momentum is 0.8. In addition, the value of optimizer weight decay is 0.0005.

Common Evaluation Indicators

The mean average precision (mAP) of all categories in the data set is often used to evaluate the effect of the object identification model (Teng et al. 2022; Inam et al. 2023). The mAP can be calculated from Eqs. (13) to (16):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{13}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{14}$$

$$AP = \int_0^1 (\text{Precision}) d(\text{Recall}) \tag{15}$$

$$\text{mAP} = \frac{\sum_{i=0}^{n} AP(i)}{n} \tag{16}$$

where Precision represents the ratio of correctly predicted damage pixels with respect to all the pixels predicted as damage pixels; Recall represents the ratio of correctly predicted damage pixels with respect to all the true damage pixels. Generally, when Precision is high, Recall is often low, and when Recall is high, Precision is often low. AP represents the area between the Precision–Recall curve and the axes of one category of damage, and mAP represents the mean value of the AP between all categories of damages. True positive (TP) is the number of pixels that are predicted to be positive and whose actual labels are positive; False positive (FP) is the number of pixels that are predicted to be positive and whose actual labels are negative; False negative (FN) is the number of pixels that are predicted to be negative and whose actual labels are positive; and n is the damage category.

Image Mosaic Method Using Identified Subimages

After the network is trained with the subimages and their label information, it is necessary to evaluate the effect of the model using the test data (Xie et al. 2018; Wang et al. 2020). However, the images in the test set are not labeled, so the images should be cropped using the traditional sliding window cropping method. Moreover, after the subimages are identified, it is necessary to restore their prediction boxes to the original high-resolution images so that inspectors can conveniently and quickly locate bridge damage. Fig. 4 shows the process of testing the multidamage high-resolution images, and the coordinates of the bounding box of the damage in the original images can be calculated according to Eq. (7).

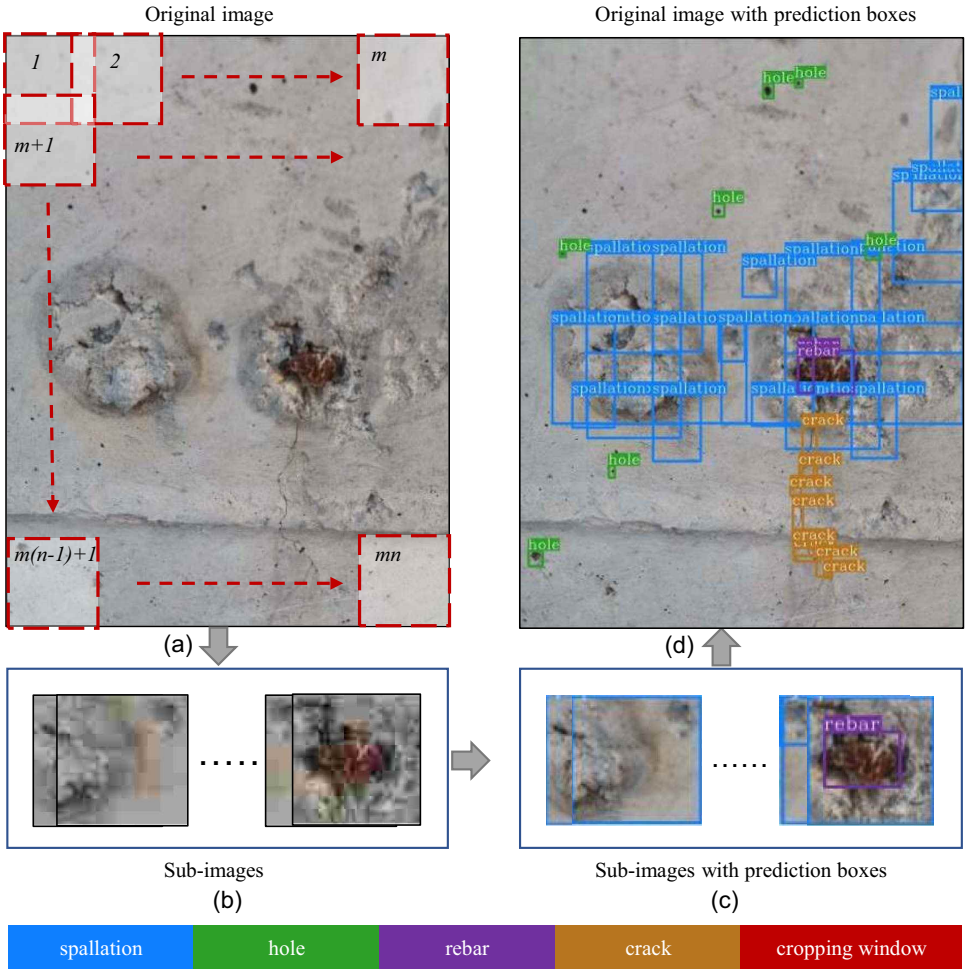

Fig. 4. Process of testing the multidamage high-resolution images: (a) the original high-resolution image; (b) the subimages cropped by the sliding window method; (c) the subimages identified by the multidamage identification model using AWCM; and (d) the high-resolution image mosaiced with the subimages identified.

J. Perform. Constr. Facil., 2024, 38(5): 04024026

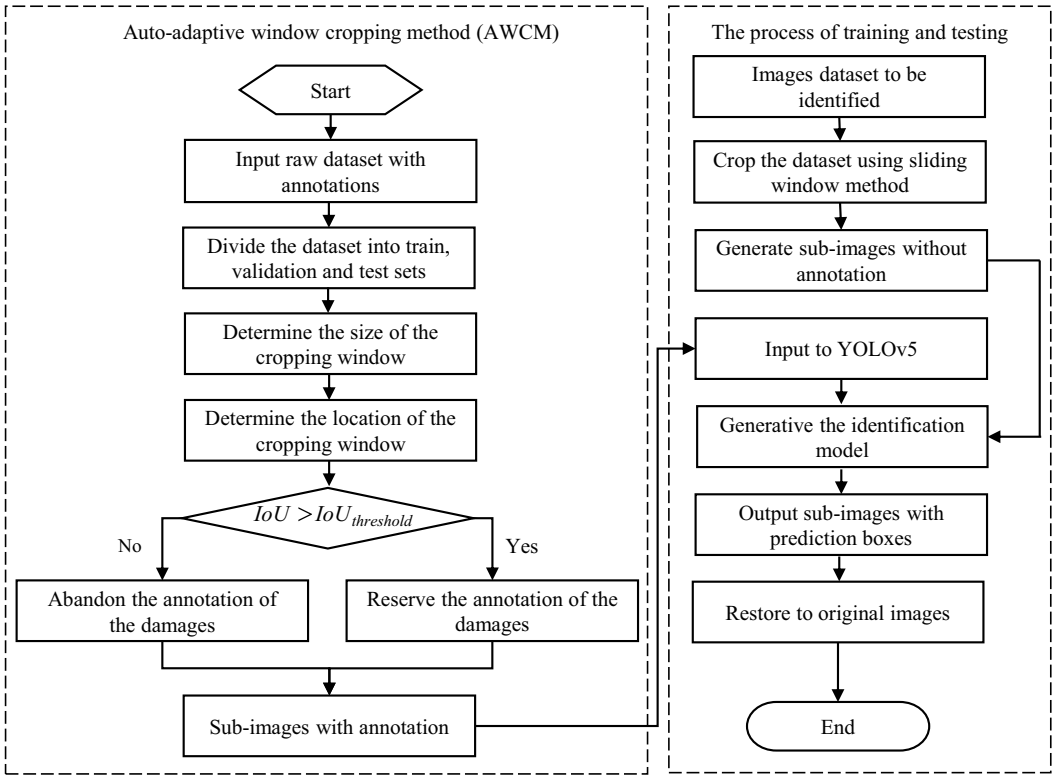

Fig. 5. Flowchart of multidamage identification using the autoadaptive window cropping method.

Finally, to avoid the situation in which there are many bounding boxes for one damage in the original high-resolution images, nonmaximum suppression (NMS) is used to find the best target boundary box and eliminate redundant boundary boxes. The entire process of the proposed framework is summarized in Fig. 5.

Multidamage Data Set of Concrete Bridge

This section describes the features of the data set that contain four different types of surface damage and then introduces the process of balancing and augmenting the data set to avoid overfitting.

Features of Data Set

All the images in the data set are captured by the authors on several reinforced concrete bridges in Nong 'an County, Jilin Province. In the data set, 1,012 images with 3,000 × 4,000 pixel resolution are captured by a smartphone under different lighting conditions and distances. There are many types of damage, including cracks, spallation, holes, rebars, efflorescence, honeycombs, and pitting surfaces, in the data set. In this paper, the four most numerous damages (cracks, spallation, holes, and rebars), which have different features and scales, are studied. The cracks are linear and small in width; the spallation is lumpy and irregular in shape; the holes are lumpy and vary in size; moreover, some rebars are striated, and others are lumpy. Fig. 6 shows some examples of the four types of damage. The damages in the data set are multiple, multiscale, and unevenly distributed in the high-resolution images. Some images only show one type of damage, while others show multiple damages simultaneously.

The ground truths of the four damage types are manually labeled using *Labelimg* software. Fig. 7 shows four examples of both original images and labeled images. Then 1,012 original high-resolution images and their corresponding annotation files are divided into training sets, verification sets, and test sets in a ratio of 8:1:1. To input images of an appropriate size into the deep learning network, the high-resolution images in training sets, verification sets, and test sets are cropped by the AWCM into many subimages. Finally, the subimages of the training set and verification set are input into the network for training and the subimages in the test set are used for testing. A basic overview of the data set before and after cropping is shown in Table 1.

Balance and Augmentation of Multidamage Data Set

From Table 1, the data set is unbalanced after cropping. After cropping, the numbers of cracks, spallation, holes and rebars in the training set are 4,424, 1,925, 40,465, and 1,261, respectively, and the numbers of cracks, spallation, holes and rebars in the validation set are 658, 251, 6,269, and 179, respectively. The most common damage type in the training set is holes (40,465), the least common damage type is rebars (1,261), and the ratio between them is nearly 32:1. Similarly, in the verification set, the most common damage type is holes (6,269), the least common damage type is rebar (179), and the ratio between the two is approximately 35:1. To avoid overfitting, it is necessary to balance the volume of different types of damage.

Data augmentation is highly important for increasing the robustness of the model. When the number of damages is small, to expand the data set, the images can be rotated, translated, scaled, and flipped. In addition, changing the color of images can also make the images different. In addition, the data set can be enlarged by introducing noise and using image filters (Leach et al. 2021). When the number of damages is large, the damages should be randomly removed to balance the data set.

For the data set in this paper, the images containing mainly holes in the training set and validation set are eliminated randomly, while

J. Perform. Constr. Facil., 2024, 38(5): 04024026

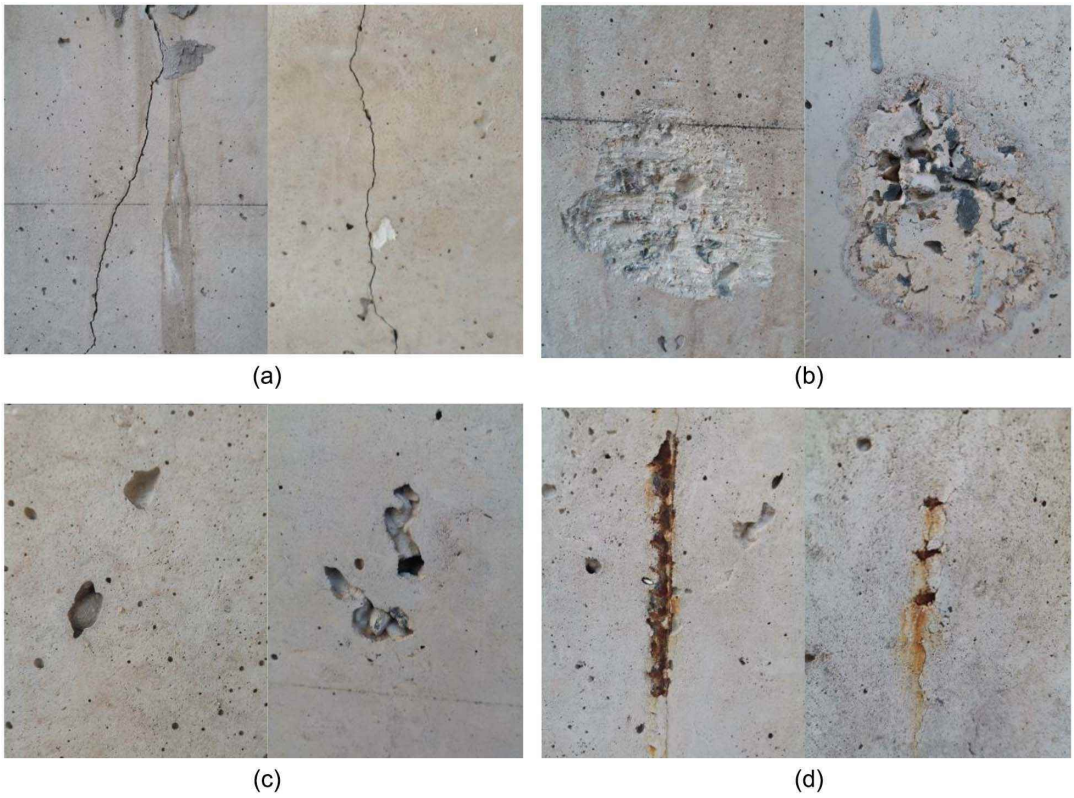

Fig. 6. Samples of four damages: (a) images with cracks; (b) images with spallation; (c) images with holes; and (d) images with rebars.

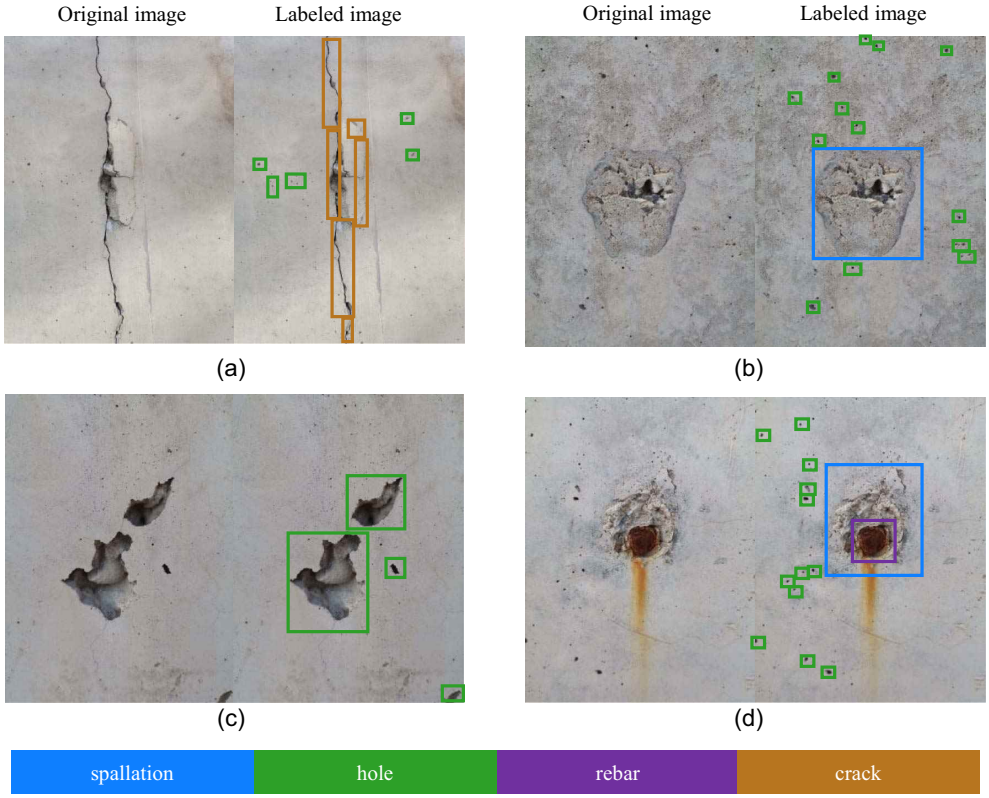

Fig. 7. Examples of both original images and labeled images: (a) an original image and a labeled image with cracks and holes; (b) an original image and a labeled image with spallation and holes; (c) an original image and a labeled image with holes; and (d) an original image and a labeled image with spallation, a rebar, and holes.

J. Perform. Constr. Facil., 2024, 38(5): 04024026

Table 1. Basic information on the data set before and after cropping

Data set name	Damage type	Before cropping: Number of images	Before cropping: Number of damages	After cropping: Number of images	After cropping: Number of damages
Training set	Crack	819	1,589	18,279	4,424
	Spallation		1,108		1,925
	Hole		15,037		40,465
	Rebar		545		1,261
Validation set	Crack	91	222	2,458	658
	Spallation		135		251
	Hole		2,028		6,269
	Rebar		73		179
Test set	Crack	102	145	1,994	341
	Spallation		155		235
	Hole		1,610		4,045
	Rebar		84		189

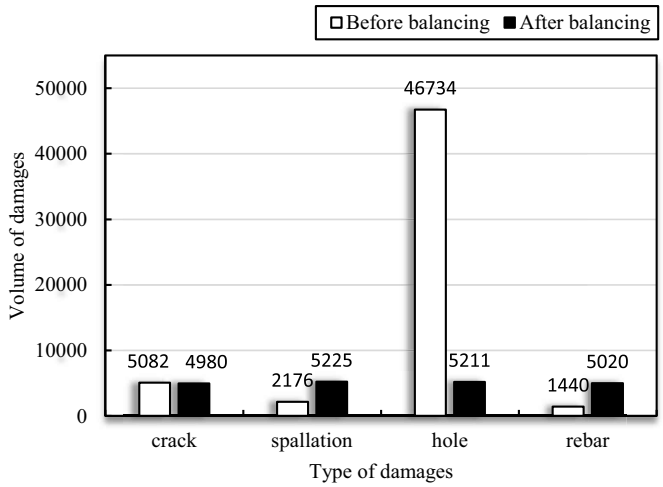

Fig. 8. Volume of different types of damage in the training and validation sets before and after balancing.

those containing mainly the other three types of damage are randomly expanded using four basic image augmentation methods. The volumes of different types of damage in the training set and validation set before and after balancing are shown in Fig. 8. However, the damages in the test set do not influence the effect of the model, so it does not need to be balanced.

The previous methods are used to balance the data set offline before the images are input into the deep learning network. Afterward, an online image augmentation method named Mosaic is used during the training process. This method crops 4 pictures at random and then stitches them into a picture as training data, which can not only enrich the background of the images but also improve the universality and robustness of the network.

Case Study Using Proposed Multidamage Identification Framework

This study mainly focuses on the identification of multiple types of damage on concrete surfaces in high-resolution imagery. To verify the identification effectiveness, three other image processing

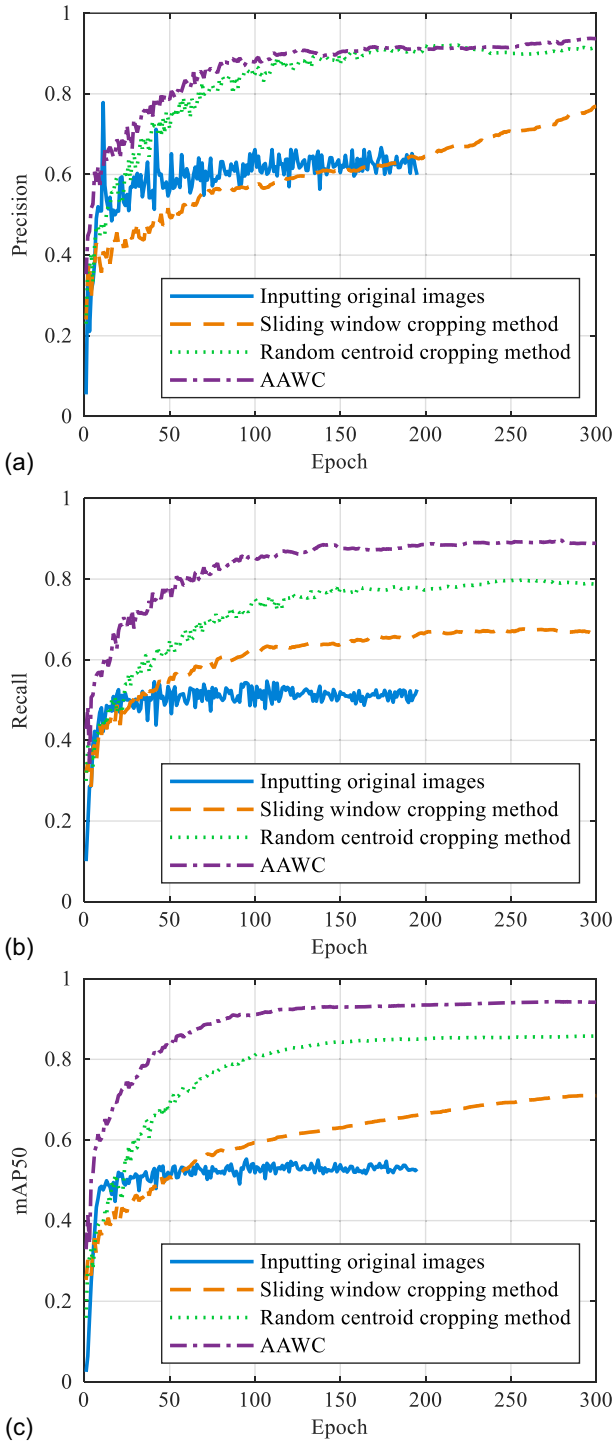

Fig. 9. Evaluation indicators change curves: (a) the change curve of precision; (b) the change curve of recall; and (c) the change curve of mAP_{50}.

methods (inputting original images directly, the sliding window cropping method, and the random centroid cropping method) are compared with the AWCM proposed in this paper.

All the methods are based on the same training, validation, and test sets and are combined with the same deep learning network, YOLOv5, which has the same hyperparameters. The change curves of the evaluation indicators are shown in Fig. 9, and the training results of the four methods are shown in Table 2.

Table 2. The mAP_{50} values of the four methods

Class	Inputting original images mAP_{50}	Sliding window cropping method mAP_{50}	Random centroid cropping method mAP_{50}	Autoadaptive window cropping method mAP_{50}
All	0.553	0.710	0.858	0.942
Crack	0.186	0.711	0.879	0.869
Spallation	0.586	0.485	0.731	0.981
Hole	0.648	0.678	0.876	0.923
Rebar	0.790	0.965	0.946	0.994

The mAP_{50} is the mean average precision when the IoU threshold between the ground truth and the prediction box is set to 50%. Specifically, if the IoU between the ground truth and the prediction box exceeds 50%, the prediction box is regarded as true; otherwise, it is regarded as false.

In Figs. 9(a–c), the three change curves of the model using AWCM increase steadily, and when they reach the stable stage, their values are all higher than those of the models using the other three image processing methods, indicating that the identification effect of the model using AWCM is better than that of any other model. The curves of the model trained on the original images all stop at epoch 195 because there is an early stopping setting in the YOLOv5 network. If the effect of the model does not change in the last 100 epochs, the training process is stopped early to save time.

From Table 2, the mAP_{50} of all classes of damage using AWCM is the highest, with a value of 94.2%, which is 38.9% higher than that achieved by inputting original images directly, 23.2% higher than the figure of the sliding window cropping method, and 8.4% higher than that of the random centroid cropping method. Moreover, the mAP_{50} of any class of damage identified by the model using AWCM is much higher than that achieved by the model trained on the original images directly and that using the sliding window cropping method. Conversely, although the mAP_{50} of cracks using AWCM is 1% lower than that using the random centroid cropping method, the mAP_{50} values of the other three damage types are much greater than those of their counterparts using the random centroid cropping method. The mAP_{50} values of cracks, holes, and rebars identified by the model using AWCM are 98.1%, 92.3%, and 99.4%, respectively, while those of cracks, holes, and rebars identified by the model using the random centroid cropping method are 73.1%, 87.6%, and 94.6%, respectively. Therefore, the effectiveness of multidamage identification using AWCM proposed in this paper is better than that of other image processing methods.

Fig. 10 shows the confusion matrices of four different image processing methods. The values on the diagonal are the ratios that the model predicted correctly, and the rest are the ratios that are missed or misidentified. The values on the diagonal of the AWCM

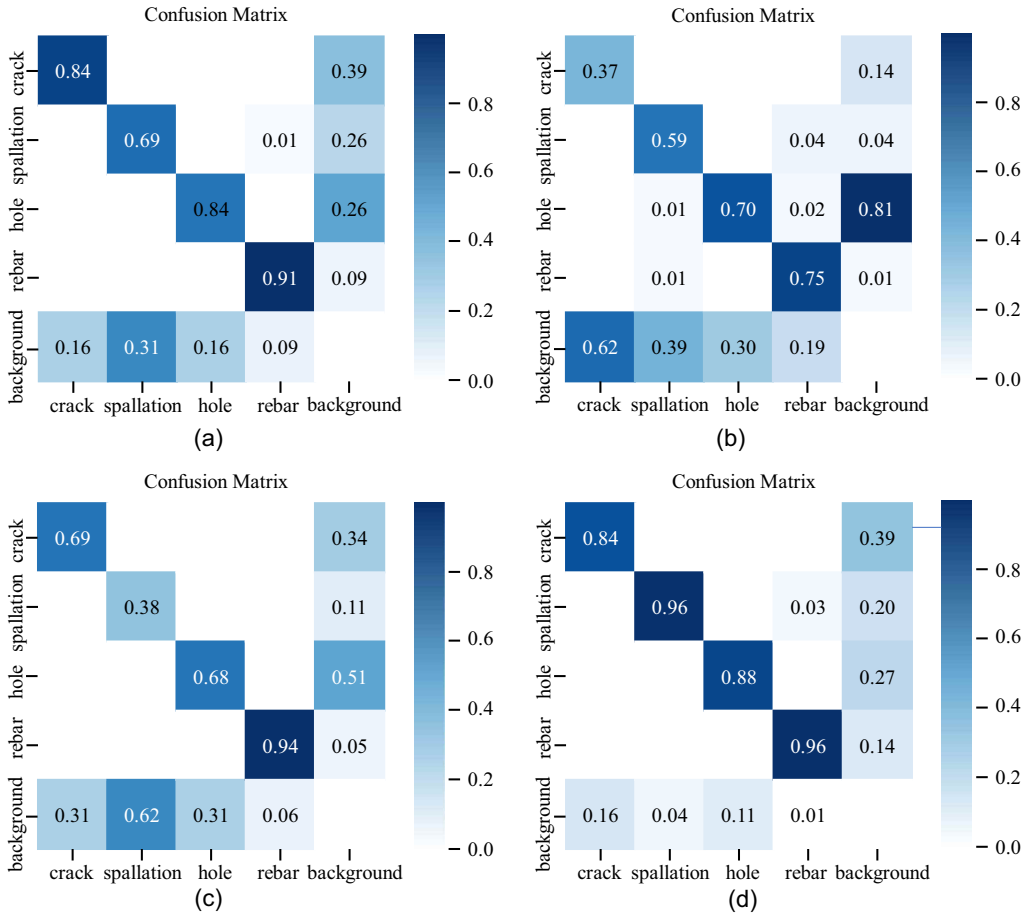

Fig. 10. Confusion matrices of four different image processing methods: (a) the confusion matrix corresponding to the inputting original images method; (b) the confusion matrix corresponding to the sliding window cropping method; (c) the confusion matrix corresponding to the random centroid cropping method; and (d) the confusion matrix corresponding to the autoadaptive window cropping method.

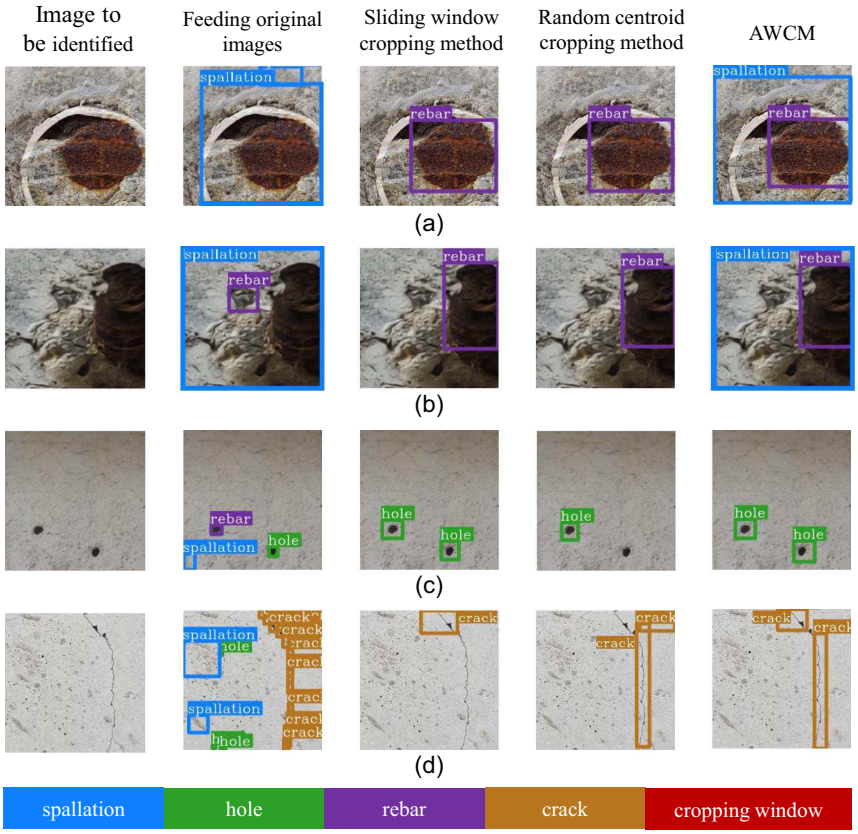

Fig. 11. Identification results of four images with different damages using four different image processing methods.

method are higher or equal to those of the other methods. Fig. 11 shows a comparison of the identification results of four different image processing methods on four images with different damage types. In Fig. 11(a), there is a rebar and a piece of spallation, which are both identified correctly by the model using the AWCM. By comparison, the model trained on the original images identifies only the spallation, and the area of the prediction box does not cover the whole spallation, while the other two models using the sliding window cropping method and random centroid cropping method identify only the rebar correctly but miss the spallation. Similarly, an image with a rebar and a piece of spallation is presented in Fig. 11(b), where the model trained on the original images fails to identify the rebar, although it correctly identifies the spallation, and the other two models combining the sliding window cropping method and random centroid cropping method ultimately identify only the rebar. Model using the AWCM performs the best and can identify both the rebar and the spallation simultaneously. In Fig. 11(c), the model trained on the original images correctly identifies the hole on the right side of the image but incorrectly identifies the other hole and the background as a rebar and a piece of spallation. Comparatively, the model combining the random centroid cropping method identifies only the hole on the left side of the image, while the other models identify both holes successfully. A thin crack is shown in Fig. 11(d), where the model trained on the original images incorrectly identifies the stains on the concrete as spallation and holes, although it correctly identifies the crack with small prediction boxes. Moreover, the two models using the sliding window cropping method and random centroid cropping method identify the upper part and lower part of the crack, respectively, and the model using the random centroid cropping method incorrectly identifies the background as a crack. The model using the AWCM can not only correctly identify the cracks but also does not identify the background as damage. Therefore, the effectiveness of the identification model using the AWCM is better than that of the other methods, and the other models have different degrees of false identification and missing identification.

After subimages identification by the model using the AWCM, the prediction boxes of the damages should be restored to the original high-resolution images. Some examples are shown in Fig. 12.

Conclusions

This paper proposes a method to identify multiple types of damage on the surface of concrete bridges in high-resolution imagery based on the YOLOv5 model. To improve the ability of the deep learning model to learn damage features, an autoadaptive window cropping method is proposed to crop high-resolution images into subimages of appropriate sizes. The main conclusions are as follows:

- A novel image cropping method named the AWCM is proposed. Compared with other methods, the method in this paper can adjust the size and location of the cropping window autoadaptively according to the labeled bounding box of damage and the input size required by the network to keep the features of damages intact. Therefore, in the subimages cropped by the AWCM, the damage feature information is more intact, so that the deep learning model can learn the feature more fully, thus improving the performance of the model.
- An image mosaic method using the identified subimages is proposed. After the subimages are identified, it is necessary to restore their bounding boxes to the original high-resolution images so that inspectors can conveniently and quickly locate

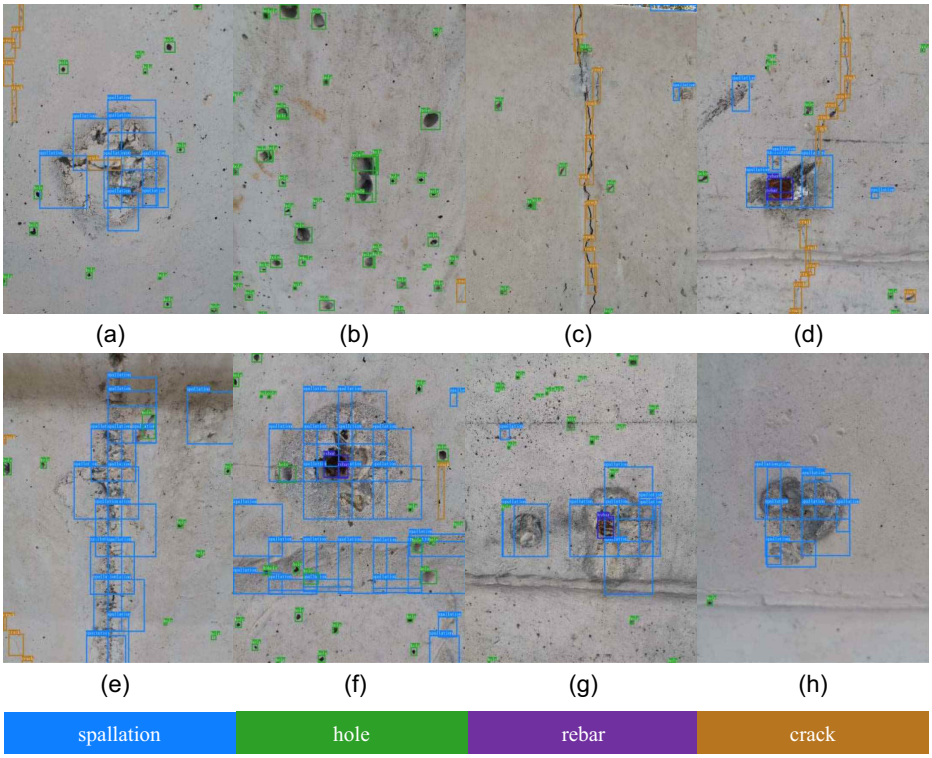

Fig. 12. Examples of restoration of the prediction box in the subimages to their original high-resolution images.

bridge surface damage. The image mosaic method can mosaic the subimages according to their corresponding location before cropping.

- A multidamage identification framework from training to testing is constructed. In the training process, the autoadaptive window cropping method is used to crop high-resolution images into subimages of appropriate sizes, and in the test process, the sliding window cropping method is used to crop the images while the image mosaic method is used to restore the prediction boxes to the original images.

Although the autoadaptive window cropping method can ensure the integrity of at least one damage in the cropping window, when there are multiple damages in the cropping window, some damages are still cropped. As a result, the cropped damage features are incomplete, which will limit the performance of the model to some extent. It is still a problem worth studying to deal with multiple damages in the cropping window. In future work, this factor will be considered to improve the performance of the model.

Data Availability Statement

All the data, models, and code that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

This research work was jointly supported by the National Natural Science Foundation of China (Grants Nos. 52250011 and 12002224), and the Fundamental Research Funds for the Central Universities (Grant Nos. DUT22ZD213 and DUT22QN235).

References

Abdel-Qader, I., O. Abudayyeh, and M. E. Kelly. 2003. "Analysis of edge-detection techniques for crack identification in bridges." *J. Comput. Civ. Eng.* 17 (4): 255–263. https://doi.org/10.1061/(ASCE)0887-3801(2003)17:4(255).

Arulalan, V., and D. Kumar. 2023. "Efficient object detection and classification approach using HTYOLOV4 and M2RFO-CNN." *Comput. Syst. Sci. Eng.* 44 (2): 1703–1717. https://doi.org/10.32604/csse.2023.026744.

Baduge, S. K., S. Thilakarathna, J. S. Perera, M. Arashpour, P. Sharafi, B. Teodosio, A. Shringi, and P. Mendis. 2022. "Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications." *Autom. Constr.* 141 (Sep): 104440. https://doi.org/10.1016/j.autcon.2022.104440.

Cao, M. T., N. M. Nguyen, K. T. Chang, X. L. Tran, and N. D. Hoang. 2021. "Automatic recognition of concrete spall using image processing and metaheuristic optimized LogitBoost classification tree." *Adv. Eng. Software* 159 (Sep): 103031. https://doi.org/10.1016/j.advengsoft.2021.103031.

Cha, Y. J., W. Choi, G. Suh, S. Mahmoudkhani, and O. Buyukozturk. 2018. "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types." *Comput.-Aided Civ. Infrastruct. Eng.* 33 (9): 731–747. https://doi.org/10.1111/mice.12334.

Chen, L., W. Chen, L. Wang, C. Zhai, X. Hu, L. Sun, Y. Tian, X. Huang, and L. Jiang. 2023. "Convolutional neural networks (CNNs)-based multi-category damage detection and recognition of high-speed rail (HSR) reinforced concrete (RC) bridges using test images." *Eng. Struct.* 276 (Feb): 115306. https://doi.org/10.1016/j.engstruct.2022.115306.

Deng, W., Y. Mou, T. Kashiwa, S. Escalera, K. Nagai, K. Nakayama, Y. Matsuo, and H. Prendinger. 2020. "Vision based pixel-level bridge structural damage detection using a link ASPP network." *Autom. Constr.* 110 (Feb): 102973. https://doi.org/10.1016/j.autcon.2019.102973.

Du, Y. L., T. H. Yi, X. J. Li, X. L. Rong, L. J. Dong, D. W. Wang, G. Yang, and Z. Leng. 2023. "Advances in intellectualization of transportation infrastructures." *Engineering* 23 (4): 64. https://doi.org/10.1016/j.eng.2023.01.011.

Fan, C. L., and Y. J. Chung. 2022. "Supervised machine learning-based detection of concrete efflorescence." *Symmetry* 14 (11): 2384. https://doi.org/10.3390/sym14112384.

Hoang, N. D., T. C. Huynh, and V. D. Tran. 2021. "Concrete spalling severity classification using image texture analysis and a novel jellyfish search pptimized machine learning approach." *Adv. Civ. Eng.* 2021: 5551555. https://doi.org/10.1155/2021/5551555.

Inam, H., N. Ul Islam, M. U. Akram, and F. Ullah. 2023. "Smart and automated infrastructure management: A deep learning approach for crack detection in bridge images." *Sustainability* 15 (3): 1866. https://doi.org/10.3390/su15031866.

Leach, S., Y. Xue, R. Sridhar, S. Paal, Z. Wang, and R. Murphy. 2021. "Data augmentation for improving deep learning models in building inspections or postdisaster evaluation." *J. Perform. Constr. Facil.* 35 (4): 04021029. https://doi.org/10.1061/(ASCE)CF.1943-5509.0001594.

Maeda, H., Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata. 2018. "Road damage detection and classification using deep neural networks with smartphone images." *Comput.-Aided Civ. Infrastruct. Eng.* 33 (12): 1127–1141. https://doi.org/10.1111/mice.12387.

Mir, B. A., T. Sasaki, K. Nakao, K. Nagae, K. Nakada, M. Mitani, T. Tsukada, N. Osada, K. Terabayashi, and M. Jindai. 2022. "Machine learning-based evaluation of the damage caused by cracks on concrete structures." *Precis. Eng.* 76 (Jul): 314–327. https://doi.org/10.1016/j.precisioneng.2022.03.016.

Mukhiddinov, M., A. B. Abdusalomov, and J. Cho. 2022. "A wildfire smoke detection system using unmanned aerial vehicle images based on the optimized YOLOv5." *Sensors* 22 (23): 9384. https://doi.org/10.3390/s22239384.

Nishikawa, T., J. Yoshida, T. Sugiyama, and Y. Fujino. 2012. "Concrete crack detection by multiple sequential image filtering." *Comput.-Aided Civ. Infrastruct. Eng.* 27 (1): 29–47. https://doi.org/10.1111/j.1467-8667.2011.00716.x.

Qiu, Z., Z. Zhao, S. Chen, J. Zeng, Y. Huang, and B. Xiang. 2022. "Application of an improved YOLOv5 algorithm in real-time detection of foreign objects by ground penetrating radar." *Remote Sens.* 14 (8): 1895. https://doi.org/10.3390/rs14081895.

Su, S. L., and T. Gong. 2020. "Bridge pavement crack detection under uneven illumination using improved PCNN algorithm." *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* XLII-3/W10: 1033–1040. https://doi.org/10.5194/isprs-archives-XLII-3-W10-1033-2020.

Sun, L., Z. Shang, Y. Xia, S. Bhowmick, and S. Nagarajaiah. 2020. "Review of bridge structural health monitoring aided by big data and artificial intelligence: From condition assessment to damage detection." *J. Struct. Eng.* 146 (5): 04020073. https://doi.org/10.1061/(ASCE)ST.1943-541X.0002535.

Teng, S., Z. Liu, and X. Li. 2022. "Improved YOLOv3-based bridge surface defect detection by combining high- and low-resolution feature images." *Buildings* 12 (8): 1225. https://doi.org/10.3390/buildings12081225.

Wan, H., L. Gao, Z. Yuan, H. Qu, Q. Sun, H. Cheng, and R. Wang. 2023. "A novel transformer model for surface damage detection and cognition of concrete bridges." *Expert Syst. Appl.* 213 (Part B): 119019. https://doi.org/10.1016/j.eswa.2022.119019.

Wang, C., X. Hou, and Y. Liu. 2021. "Three-dimensional crack recognition by unsupervised machine learning." *Rock Mech. Rock Eng.* 54 (2): 893–903. https://doi.org/10.1007/s00603-020-02287-w.

Wang, D., M. Zhang, D. Sheng, and W. Chen. 2023. "Bolt positioning detection based on improved YOLOv5 for bridge structural health monitoring." *Sensors* 23 (1): 396. https://doi.org/10.3390/s23010396.

Wang, H. F., L. Zhai, H. Huang, L. M. Guan, K. N. Mu, and G. Wang. 2020. "Measurement for cracks at the bottom of bridges based on tethered creeping unmanned aerial vehicle." *Autom. Constr.* 119 (Nov): 103330. https://doi.org/10.1016/j.autcon.2020.103330.

Wang, W., A. Zhang, K. C. P. Wang, A. F. Braham, and S. Qiu. 2018. "Pavement crack width measurement based on Laplace's equation for continuity and unambiguity." *Comput.-Aided Civ. Infrastruct. Eng.* 33 (2): 110–123. https://doi.org/10.1111/mice.12319.

Wang, Y., J. Y. Zhang, J. X. Liu, Y. Zhang, Z. P. Chen, C. G. Li, K. He, and R. B. Yan. 2019. "Research on crack detection algorithm of the concrete bridge based on image processing." *Procedia Comput. Sci.* 154 (Jan): 610–616. https://doi.org/10.1016/j.procs.2019.06.096.

Wu, F., J. Duan, P. Ai, Z. Chen, Z. Yang, and X. Zou. 2022. "Rachis detection and three-dimensional localization of cut off point for vision-based banana robot." *Comput. Electron. Agric.* 198 (Jul): 107079. https://doi.org/10.1016/j.compag.2022.107079.

Xie, R., J. Yao, K. Liu, X. Lu, Y. Liu, M. Xia, and Q. Zeng. 2018. "Automatic multi-image stitching for concrete bridge inspection by combining point and line features." *Autom. Constr.* 90 (Jun): 265–280. https://doi.org/10.1016/j.autcon.2018.02.021.

Xu, Y., and J. M. W. Brownjohn. 2018. "Review of machine-vision based methodologies for displacement measurement in civil structures." *J. Civ. Struct. Health Monit.* 8 (1): 91–110. https://doi.org/10.1007/s13349-017-0261-4.

Xu, Y., S. Wei, Y. Bao, and H. Li. 2019. "Automatic seismic damage identification of reinforced concrete columns from images by a region-based deep convolutional neural network." *Struct. Control. Health Monit.* 26 (3): e2313. https://doi.org/10.1002/stc.2313.

Yu, J. C., T. H. Yi, S. H. Zhang, H. N. Li, Y. F. Wang, and X. D. Mei. 2023. "Automatic quantitative identification of bridge surface cracks based on deep learning." *J. Perform. Constr. Facil.* 37 (1): 04022072. https://doi.org/10.1061/JPCFEV.CFENG-4238.

Yu, L., S. He, X. Liu, S. Jiang, and S. Xiang. 2022. "Intelligent crack detection and quantification in the concrete bridge: A deep learning-assisted image processing approach." *Adv. Civ. Eng.* 2022: 1813821. https://doi.org/10.1155/2022/1813821.

Zhang, C., C. Chang, and M. Jamshidi. 2020. "Concrete bridge surface damage detection using a single-stage detector." *Comput.-Aided Civ. Infrastruct. Eng.* 35 (4): 389–409. https://doi.org/10.1111/mice.12500.

Zhang, Y. C., T. H. Yi, S. Lin, H. N. Li, and S. Lv. 2022. "Automatic corrosive environment detection of RC bridge decks from ground-penetrating radar data based on deep learning." *J. Perform. Constr. Facil.* 36 (2): 04022011. https://doi.org/10.1061/(ASCE)CF.1943-5509.0001712.

Zhao, S., F. Kang, and J. Li. 2022. "Concrete dam damage detection and localisation based on YOLOv5s-HSC and photogrammetric 3D reconstruction." *Autom. Constr.* 143 (Nov): 104555. https://doi.org/10.1016/j.autcon.2022.104555.

Zhao, Z., T. Liu, and X. Zhao. 2021. "Variable selection from image texture feature for automatic classification of concrete surface voids." *Comput. Intell. Neurosci.* 2021: 5538573. https://doi.org/10.1155/2021/5538573.