Measurement Science
and Technology

PAPER

Three-dimensional reconstruction and damage localization of bridge undersides based on close-range photography using UAV

To cite this article: Shang Jiang *et al* 2025 *Meas. Sci. Technol.* **36** 015423

IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **36** (2025) 015423 (18pp) https://doi.org/10.1088/1361-6501/ad90fb

Three-dimensional reconstruction and damage localization of bridge undersides based on close-range photography using UAV

Shang Jiang[1,2,*], **Yufeng Zhang**[3,4], **Feiyu Wang**[2] and **Yichao Xu**[3,4]

[1] School of Transportation and Civil Engineering, Nantong University, Nantong, People's Republic of China
[2] School of Civil Engineering, Southeast University, Nanjing, People's Republic of China
[3] Jiangsu Transportation Institute Group, Nanjing, People's Republic of China
[4] National Key Laboratory of Safety, Durability and Healthy Operation of Long Span Bridges, Nanjing, People's Republic of China

E-mail: shangjiang@ntu.edu.cn

Received 11 June 2024, revised 11 October 2024
Accepted for publication 11 November 2024
Published 20 November 2024

Abstract

Damage inspection on the undersides of bridges is an important and challenging part of routine bridge inspections. A method for 3D reconstruction and damage localization of bridge undersides based on close-range photography by unmanned aerial vehicle (UAV) and stereo vision combined with deep learning algorithms is proposed, the specific contributions include: (1) proposing a close-range photography method for acquiring high-resolution images from multiple perspectives of the bridge underside by UAVs, serving as the data source for damage analysis; (2) applying a deep learning-assisted segmentation method to optimize the multi-view geometry-based 3D reconstruction method, improving the efficiency of three-dimensional reconstruction, and defining the projection direction from the 3D reconstruction results to obtain ultra-high-resolution panoramic images of the bridge underside; (3) addressing the issue of detecting minor damages in large panoramic images by using a slice-assisted reasoning module and a lightweight convolutional YOLO v8 network to identify exposed steel bars corroded due to concrete damage in the panoramic images, and defining a coordinate system to localize the damages on the bridge underside. The proposed method was applied to damage detection and localization on the underside of a 160 m span main span of an in-service concrete bridge. The results demonstrate that the proposed method can quickly and accurately identify exposed steel bar corrosion on the bridge underside and output reports, proving the practicality of the proposed method.

Keywords: bridge inspection, damage detection, aerial photography, 3D reconstruction, deep learning

* Author to whom any correspondence should be addressed.

1. Introduction

Bridges are the lifelines of transportation networks. Once a bridge is damaged or even collapses, it not only paralyzes local traffic but also leads to serious accidents and significant economic losses. Bridges, subjected to long-term traffic loads and unexpected events during their service life, inevitably sustain damage, gradually reducing their durability and load-bearing capacity, eventually affecting their safe operation. Regularly inspecting the health status of bridges and repairing or strengthening them according to the damage situation is an important means to maintain bridge safety [1]. Currently, widely used manual bridge inspection methods are gradually unable to meet the extensive bridge inspection demand due to factors such as low efficiency, high costs, and high risks. Therefore, how to maintain the health and serviceability of a large number of bridges has become a research focus in the field of civil engineering [2]. In recent years, with the rapid development of unmanned aerial vehicles (UAVs) and robotics technology, methods for detecting bridge surface defects using these flexible automated platforms, especially for areas difficult for personnel to reach, have shown great potential [3–5]. Additionally, visual measurement methods based on computer vision technology have gradually improved and have been proven to be suitable for rapidly and automatically analyzing defects from massive image data [6–8]. Based on this, this paper proposes to study the possibility of using the aforementioned technologies to detect defects on bridge surfaces, especially at the bottom of bridges, with the aim of achieving rapid and automated identification and localization of bottom bridge defects.

For the detection of surface defects on bridges, the bottom of bridges is undoubtedly a key area of focus and is also an area difficult for personnel to reach during inspection. For concrete bridges, which constitute the largest proportion of bridges, cracks and concrete spalling at the bottom occur most frequently due to tensile forces acting on the bottom under loading conditions. Generally, these fine cracks and minor spalling do not affect the overall safety performance of bridges. However, due to crack propagation and spalling, concrete reinforcement is exposed and corroded, reducing the effective area of reinforcement over long-term service, potentially leading to deterioration of the bridge's durability and eventually causing local damage. Therefore, for concrete bridges, regularly inspecting the exposure and corrosion of reinforcement at the bottom of bridges and repairing the concrete at the bottom according to the inspection results are important tasks in daily bridge management [9]. Research and development of UAV and robotic platforms suitable for image acquisition at the bottom of bridges to achieve high-precision image acquisition have been a recent research hotspot in bridge inspection. For UAV platforms, although they can quickly reach the bottom of bridges through flight, commonly used UAVs generally rely on GPS satellite signals to provide location data. However, the bottom of bridges is often shielded from GPS signals by the beam structure, resulting in weak GPS signals. This issue has once limited the application of UAVs in defect detection at the bottom of bridges. However, in recent years, with the development and maturation of localization technology, local positioning methods based on stereo vision [10], LiDAR [11], ultrasonic beacons [12], and ultra wide band (UWB) beacons [13] have been proposed and applied in UAV positioning in GPS-denied environments, enabling UAVs to fly normally in environments such as the bottom of bridges. For example, Wang *et al* proposed a navigation system based on stereo vision to assist UAV systems in autonomously completing takeoff, localization, navigation, and landing in unknown environments [14]. Kang and Cha proposed an ultrasonic beacon system for UAV systems to provide autonomous driving in GPS-deprived environments, achieving decimeter-level positioning accuracy [12]. Jiang *et al* developed a UAV for defect detection at locations without GPS, such as the bottom of bridges, using the proposed binocular vision and inertial navigation fusion visual odometer algorithm for UAV positioning [10]. They further developed a collision-capable UAV using UWB beacon positioning to enable UAV flight and detection at the bottom of steel structure bridges [13]. Meanwhile, some climbing robots developed based on principles such as magnetic suction and vacuum adsorption have gradually been applied to bridge inspection, making the acquisition of images at the bottom of bridges potentially unmanned and automated [15, 16]. For example, Wang *et al* developed a magnetic suction-based crawler inspection robot, consisting of two traction modules and one connecting module. Each traction module is equipped with magnetic adhesion and crawling traction sub-modules, and the two traction modules are connected by a linking module with four degrees of passive compliance, enabling the robot to crawl adaptively on surfaces with different curvatures [15]. Nguyen *et al* developed a variable climbing robot [17], which consists of two independent drive modules, and the adhesion force of each drive module is provided by magnetic wheels that can move up and down, thereby enabling the robot to move at bending nodes. In this context, considering the task of 3D reconstruction of bridges and the accuracy of defect detection, researching specific shooting methods for image acquisition is the focus of bottom bridge defect data collection.

After obtaining image data of the bottom of bridges using platforms such as UAVs, automating the analysis of defects from the data is also a widely studied focus in defect detection. Methods based on image processing and deep learning for defect analysis are commonly adopted in most existing research. Currently, most research focuses on designing deep learning networks or improving existing ones to maintain good stability and high accuracy in defect detection [18]. For example, Zhang *et al* proposed a CrackNet network, which features no pooling layers and thus predicts each pixel separately, achieving pixel-level recognition accuracy [7]. Ni *et al* proposed a dual-scale detection method based on CNN for detecting cracks of two scales-wide and fine, and introduced a sub-pixel crack width measurement method based on Zernike moments to detect the width of fine cracks, enabling accurate

detection of cracks with pixels less than 5 [19]. Xu *et al* proposed a few-shot meta-learning method based on nested attributes for recognizing 10 representative types of defects, overcoming the low robustness issue of traditional networks on limited datasets [20]. These studies have contributed numerous algorithms and networks, laying the foundation for the analysis of bridge defect images. However, for bridge management authorities, obtaining numerous discrete defect identification images alone still cannot meet the requirements. Locating the position of defects on the bridge surface from the images is equally important. In recent years, some scholars have also conducted research on how to derive the real positions of defects from images acquired by platforms such as UAVs. One approach is to use the positioning information of platforms such as UAVs themselves to calculate the position of defects. For example, Kang and Cha used positioning data from ultrasonic beacons on UAVs to determine the position of defects [12]. Yoon *et al* used distance information between the camera and the target surface to calculate the coordinates of the center of each captured image, thus determining the coordinates of features in the image [21]. This method relies on the platform's own positioning, which is difficult to reflect on the bridge structure itself due to the inconsistency of coordinate systems. Another approach is to use image stitching to obtain panoramic images of local structures, thereby marking the positions of defects in the panoramic images for defect localization in local areas. For example, Jiang and Zhang applied stitching methods in their research on climbing UAVs to obtain panoramic images of the structure surface, enabling direct observation of identified cracks from the panoramic images [22]. Attard *et al* also used image stitching technology in surface detection within tunnels, obtaining panoramic images of tunnel surfaces containing defects [23]. This method is suitable for local areas, but for large-span bridges spanning hundreds of meters, stitching panoramic images of the bottom with millimeter-level accuracy for defect identification poses a significant challenge and requires further research.

Based on the above review, this study proposes a method for automatic identification and localization of bottom bridge defects based on close-range photography by UAVs and visual analysis algorithms. Using image data of the bottom of bridges obtained by UAVs at close range, a 3D model of the bottom of the bridge is reconstructed using a multi-view geometry-based method. Subsequently, an ultra-large-size panoramic image of the bottom is projected, and deep learning networks are utilized to identify exposed reinforcement and corrosion defects in the panoramic image. The innovation of this method lies in: (1) proposing a method of bottom bridge image acquisition based on close-range photography by UAVs, which achieves defect identification accuracy while also being suitable for 3D reconstruction; (2) applying deep learning instance segmentation methods to generate masks of source images, focusing the 3D reconstruction process only on the bridge itself, thereby obtaining a bridge 3D model free of background clutter more efficiently, facilitating the acquisition of ultra-high-resolution panoramic images of the bottom using projection; (3) for the analysis of ultra-large-size panoramic images, proposing the use of a slice-assisted inference module and a lightweight convolutional YOLO v8 network to identify defects, and establishing a coordinate system from the identified defects in the panoramic image to obtain defect localization data.

The remaining sections of this paper are arranged as follows: section 2 introduces the proposed method framework and elaborates on its core methods and technologies in detail; section 3 presents the application and validation of the proposed method on an in-service bridge with a main span of 160 m; section 4 provides the conclusion of this paper.

2. Framework for bridge panoramic image generation and damage detection methods

This section introduces the proposed method framework, as illustrated in figure 1. The proposed method consists of two parts: data acquisition and analysis. For data acquisition, this method uses UAVs to capture images of the bottom of bridges. During the capture process, key considerations include setting the camera's shooting angle, image overlap rate, and the distance between the camera and the bottom of the bridge. In data analysis, depending on the goals of the analysis, it can be divided into two key components: generating ultra-high-resolution panoramic images of the bottom of bridges based on 3D reconstruction and projection, and identifying and locating defects in panoramic images based on deep learning. For the first part of panoramic image generation, this section adopts a 3D reconstruction method based on multi-view geometry to calculate the bridge's 3D point cloud, overlaying texture from the source images, and then uses a projection method from the bottom to obtain ultra-high-resolution panoramic orthoimages of the bottom of the bridge. For the second part of defect identification and localization, the core is proposing a YOLO v8 network improved with slice-assisted inference module and lightweight convolution as the defect identification model. A dataset of defect images from a large number of in-service concrete bridges is collected and used for training the proposed model, ensuring high stability and accuracy required for defect detection. Finally, the proposed network is applied to ultra-large-scale panoramic images for defect identification, and a coordinate system is established from the identified defects in the panoramic image to obtain defect location information.

2.1. Close-range photography method for UAV at the bottom of bridges

When using UAVs to obtain images of the bottom of bridges for defect detection, it is necessary to first determine the appropriate UAV and camera for the task. Since GPS signals are weak at the bottom of bridges, UAVs should have local positioning capabilities. Additionally, the bottom of bridges generally has many obstacles such as piers and surrounding trees, so UAVs should have obstacle detection capabilities. For defect collection on bridges, UAVs should maintain a certain safety

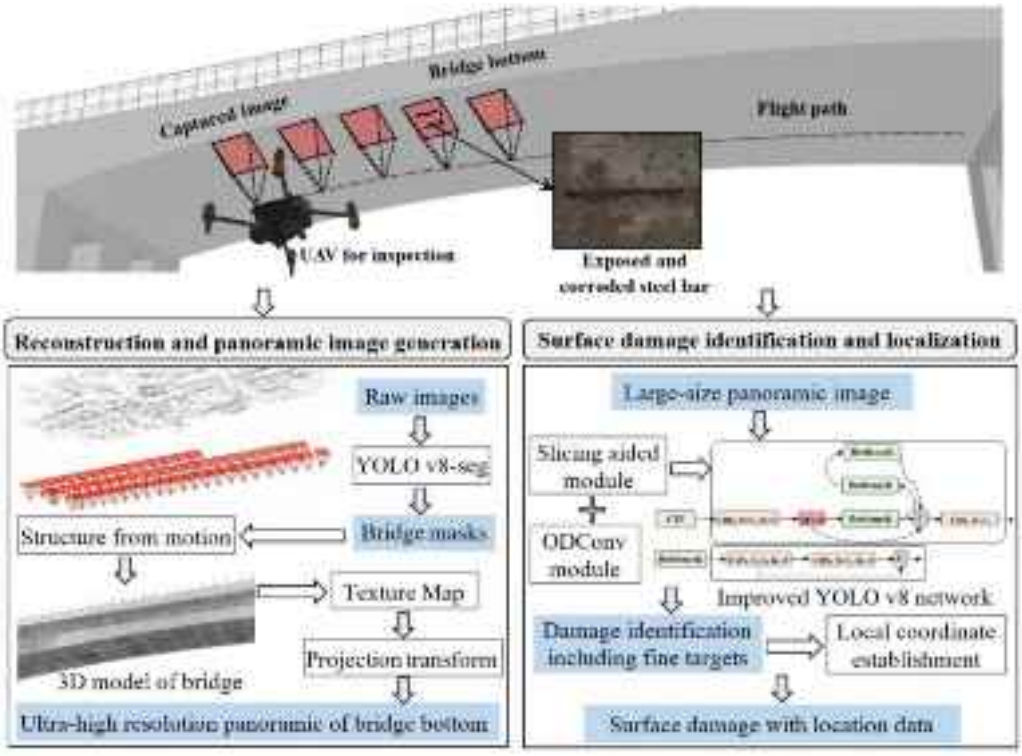


Figure 1. Framework of the proposed method.

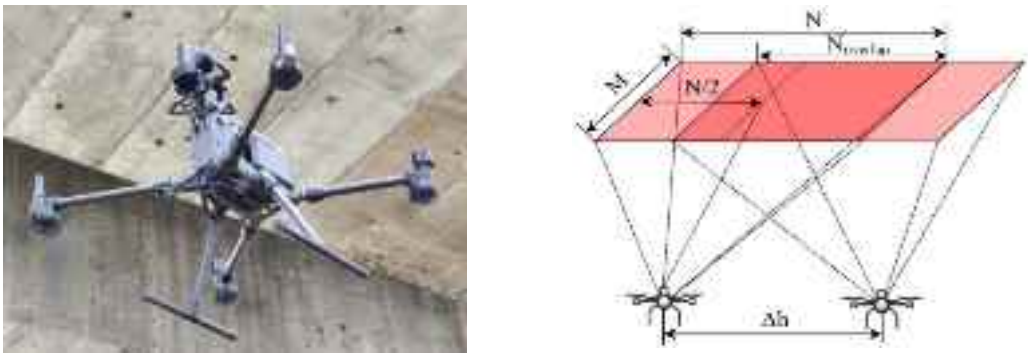
distance from the bridge bottom during flight, hence the camera used should have an appropriate focal length and high resolution. Furthermore, since UAVs need to take photos from below the bridge, the camera should be mountable on the top of the UAV to achieve a bottom-up perspective.

Based on these requirements, this study uses DJI M300RTK as the UAV for defect detection, and mapping camera DJI P-1 for image acquisition. The camera is fixed on the top of the UAV using an upward mounting bracket, enabling it to capture photos from below. The M300RTK has a maximum take-off weight of 9 kg, with a positioning accuracy of up to 0.1 m when using dynamic differential GPS positioning. It can sense obstacles in six directions: front, rear, left, right, up, and down, and provides more than 40 min of flight time. The P-1 camera used has 45 million effective pixels, with a sensor size of 35.9 $\times$ 24 mm and a lens with a 35 mm focal length, resulting in photos of size 8192 $\times$ 5460 pixels, with a minimum shooting interval of 0.7 s. The UAV and camera used in this study are shown in figure 2(a).

The type of bridge defect targeted in this study is steel reinforcement exposure and corrosion caused by concrete spalling and crack expansion. For defect measurement, an accuracy of 1 mm is generally required to accurately measure the size of the defect. During UAV flight detection, the minimum allowable distance of the upward-looking ranging sensor is 2 m. Considering possible variations in bridge cross-sectional dimensions and the presence of ancillary components such as drainage pipes at the bottom of the bridge, a safety distance of 5 m from the bridge bottom is maintained during UAV flight detection, resulting in a camera object distance of 5 m. According to the camera's pinhole imaging model [24], it can be calculated that in the images obtained under this condition, one pixel represents a real length of 0.63 mm, resulting in an image resolution of 0.63 mm, meeting the requirements of defect detection of this kind. For the defect localization required in this study, a 3D model of the bridge needs to be reconstructed from the images captured by the UAV. Therefore, there needs to be a certain overlap rate between images to meet the matching requirements of multi-view geometry. Referring to the flight overlap rate in oblique photography, which is generally a minimum of 70%, considering the high resolution of the camera used, this section calculates based on a 70% overlap rate. From the schematic diagram shown in figure 2(b), it can be derived that when the UAV captures two images, the time is 0.7 s. During this 0.7 s, the distance the UAV needs to move forward is Δh, and it needs to satisfy the following equation:

$$\Delta h = N - N_{\text{overlap}} = N - R_{\text{overlap}} k N_{\text{pixel}} \tag{1}$$

where M and N represent the size of the camera's field of view covered during the image acquisition process, determined by the camera's imaging model and parameters such as the focal length f and sensor size $M_{\text{sensor}} \times N_{\text{sensor}}$. R_{overlap} is the overlap rate of the images, set at 0.7. k is the scaling parameter for the images, previously calculated as 0.63 mm/pixel. $M_{\text{pixel}} \times N_{\text{pixel}}$ represents the pixel dimensions of the captured images. Substituting these camera parameters, the calculation yields $\Delta h = 1.03$ m, with a camera capture interval of 0.7 s,



(a) The applied UAV (b) Image capture with overlap ratio

Figure 2. The employed UAV (a), and overlapped image capture (b).

indicating that the UAV's inspection flight speed does not exceed 1.47 m s^{-1}.

After determining the shooting object distance and flight speed of the UAV during inspection, it is also necessary to determine the type of flight route for the UAV inspection and the tilt angle of the camera during shooting. The flight route planning method in this section refers to the flight route planning method widely used in oblique photography methods applied in geographic information surveying. For image capture at the bottom of bridges, three flight routes are used to cover the range of the bottom of the bridge. These include several parallel flight routes distributed directly below the bridge, with the camera shooting vertically at a 90° angle from bottom to top, as well as two oblique flight routes on the left and right sides of the bridge, with the camera facing the direction of the bridge and tilted at a 45° angle, as shown in figure 3. The 90° image provides enriched texture and high resolution for the reconstruction of the model, and the 45° image facilitates the computation of the point cloud of the model's elevation and the addition of locations such as the sides of the box girders. These three flight routes can cover the entire range of the bottom of the bridge while also considering variations in the elevation, facilitating the subsequent 3D reconstruction work from the images.

With such a shooting method, when capturing the bottom of a bridge that is 10 m wide and 60 m long, each of the three flight routes requires several parallel back-and-forth flight lines, resulting in a flight distance of approximately 180 m for each flight route. Complete coverage of the bottom of the bridge requires at least 6.1 min.

2.2. Generation of bridge bottom panorama based on 3D reconstruction

This section introduces the method of generating a super-high-resolution panorama of the bridge bottom from the images obtained by the UAV with overlapping rates. This panorama will serve as the basis for damage identification and localization. Therefore, it not only needs to have sufficiently high resolution but also needs to reflect the real dimensions of the structure to provide data for damage localization. Thus, the panorama not only needs to be orthorectified, meaning there are no distortions in the image, but also needs to have realistic scale parameters, meaning each pixel represents a real-world dimension. Based on these considerations, the method used in this section is a 3D reconstruction-based approach. The 3D model of the bridge bottom established by this method has the real-scale dimensions of the bridge, so the orthorectified image of the bridge bottom projected from the 3D model can serve as the basis for damage localization.

The method used in this section can be divided into three main steps: deep learning-based source image segmentation, 3D reconstruction based on structure-from-motion (SFM), and generation of panorama based on 3D model projection, as shown in figure 4. In the first step, the YOLO v8-seg network, which is state of the art, is used for image segmentation to separate the bridge portion from the source images and generate masks for the bridge objects. Based on the segmented images, the 3D reconstruction process can be conducted. This approach not only reduces data volume and computational time but also yields clean 3D point clouds of the bridge, eliminating the need for manual segmentation and removal of background point clouds. In the second step, a widely researched and applied SFM method is applied to reconstruct 3D point clouds from multi-view images and obtain a textured 3D model of the bridge bottom through meshing and texture projection. In the third step, a projection-based orthorectified image generation method is directly applied to generate a high-resolution panorama of the bridge bottom from the textured 3D model of the bridge.

In the first part of source image segmentation, the core of the method is to establish a bridge segmentation dataset based on UAV aerial images. An appropriate instance segmentation network is selected and trained on the established dataset to obtain a high-precision bridge object segmentation network for performing source image segmentation tasks. For the establishment of the bridge segmentation dataset, images captured by UAVs of three types of bridges were collected from multiple bridge inspections, totaling 600 images. The dataset

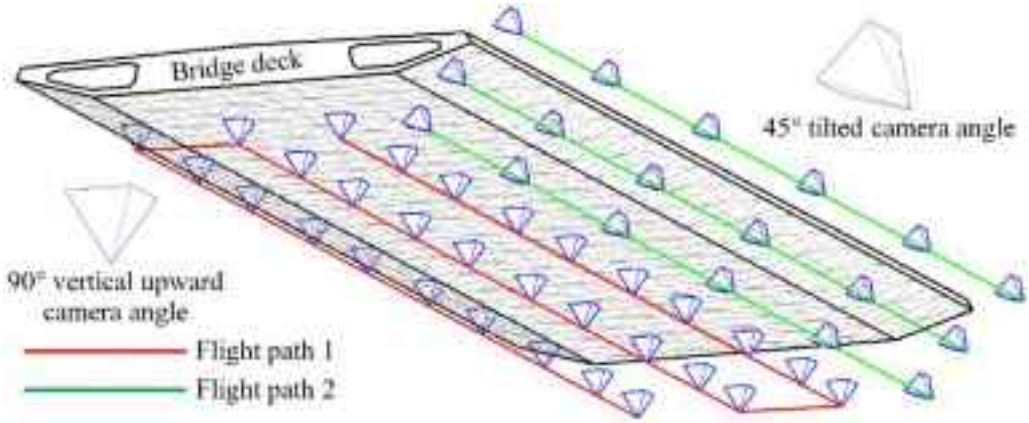


Figure 3. Approach to photographing the bottom of bridges.

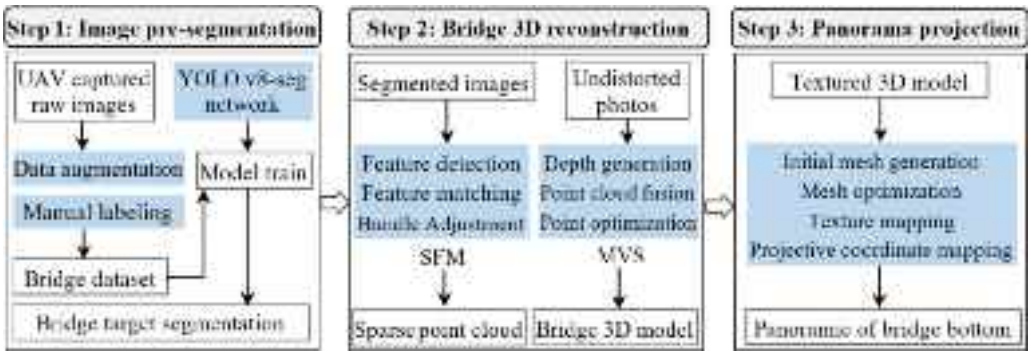
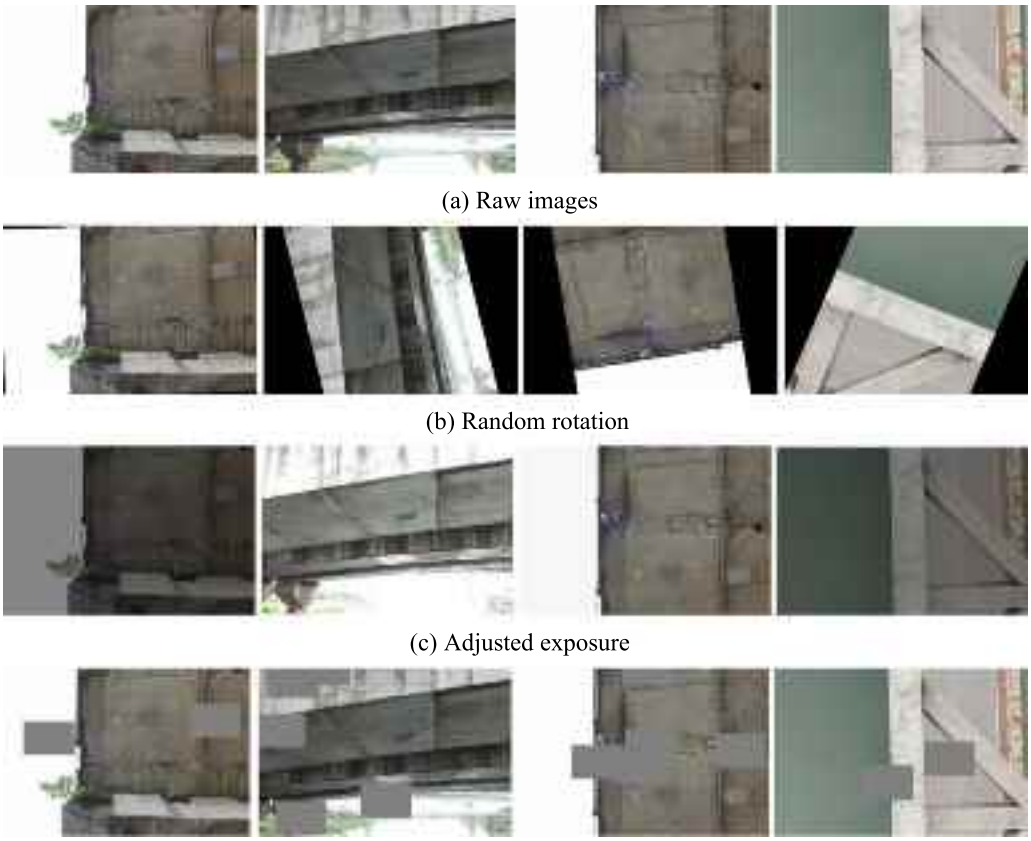


Figure 4. Steps of the method for generating a panoramic of bridge bottom.

was created using manual segmentation, and then expanded to 2400 images through image geometric transformations, addition of noise, addition of grid masks, and adjustment of exposure using image enhancement methods. The source images from three bridges and the augmented images are shown in figure 5.

After establishing the bridge segmentation dataset, selecting an appropriate instance segmentation network to perform the bridge object segmentation task in the source images is crucial for the first step shown in figure 4. After experiencing several years of rapid development, traditional deep learning networks for tasks such as image classification, object detection, and instance segmentation have seen significant improvements in both speed and accuracy. For segmentation of everyday objects, state-of-the-art networks are already capable of performing this task effectively. Since bridge object segmentation involves relatively simple backgrounds in the images and does not pose significant difficulty, this section does not involve research or improvement of instance segmentation networks but instead directly adopts the latest network to execute the task.

Automated image segmentation is a significant research focus in the field of computer vision and represents a crucial category of deep learning networks for image processing. Numerous networks tailored for image segmentation have been designed and proposed, primarily falling into two categories: semantic segmentation networks and instance segmentation networks. The former performs pixel-level segmentation of objects into different categories, distinguishing only between target and background classes, while the latter detect and delineate each individual object of interest in the image, segmenting and distinguishing each instance. Early image segmentation networks primarily relied on predicting and outputting pixel information across the entire image. Representative networks include fully convolutional network, ParseNet and U-Net networks [25–27]. These networks incorporate skip connections to merge downsampled information into the upsampled output, allowing them to extract more information from fewer labeled images, a feature extensively utilized in medical image segmentation. Regarding the later emergence of instance segmentation networks, early networks were essentially extensions of Faster R-CNN, such as Mask R-CNN, which added a module for computing binary masks based on it to achieve object segmentation [28]. Following this, various types of attention mechanism modules became popular and were incorporated into networks, enabling the models to adaptively learn the weights of feature maps. This mechanism is superior to conventional pooling methods, allowing the models to assess the importance of features at different positions and scales, such as various U-Net networks and



(a) Raw images

(b) Random rotation

(c) Adjusted exposure

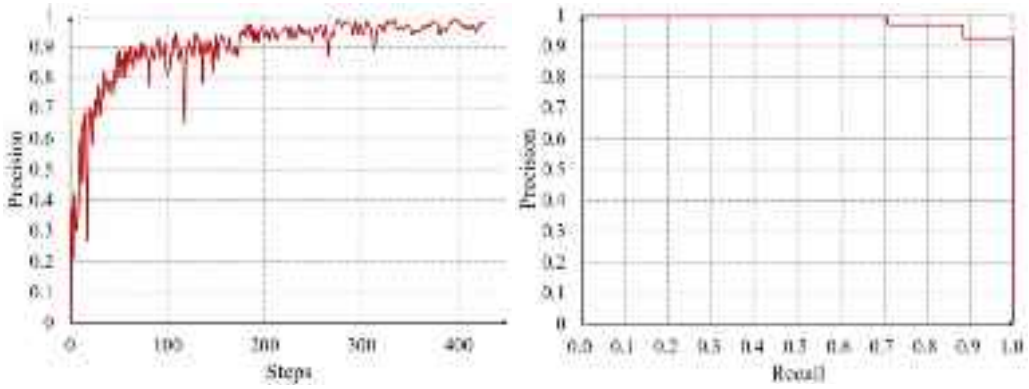
(d) Random mesh occlusion

Figure 5. The established bridge segmentation dataset.

reverse attention network improved by incorporating attention mechanisms [29]. For semantic segmentation networks, since they need to predict every pixel in the image, these networks generally have smaller input sizes and longer training and inference times. While they have advantages in segmenting thin structures, they may encounter inaccuracies in predicting pixels for large images and block-like objects. Therefore, instance segmentation networks are adopted in this section. From figure 5, it can be observed that the bridge objects to be segmented exhibit clear differences in features from the background. Hence, the requirements for the segmentation network are not stringent. This section primarily focuses on selecting smaller-sized networks to achieve faster training and inference speeds. In response to this demand, the YOLO series models, widely applied in the industry due to their fast speed and high accuracy, can be utilized for bridge object segmentation in this section [30]. Among these networks, YOLO v8 is the latest version, which optimizes the backbone, replaces the earlier C3 structure with the C2F structure, and utilizes anchor-free modules and a new loss function, distribution focal loss [31]. Therefore, this section attempts to employ the YOLO v8-seg network as the bridge object segmentation network, training and testing it on the established dataset.

The network training and testing are conducted on a computer equipped with an Nvidia RTX3090 GPU using the Ultralytics framework. The parameters set during the network training process include: 1000 training steps, a batch size of 16, an image size of 640 $\times$ 640, and a base learning rate of 0.01. The curves of the network parameters recorded during the training process are shown in figure 6. Since the YOLO v8-seg network contains networks of five different sizes, denoted as $n \sim x$, we initially tested the smallest-sized YOLO v8n-seg during training. The results showed a segmentation precision of 0.995, meeting the segmentation requirements of this section. Therefore, we directly use this trained network for segmentation of the original images.

The next step involves reconstructing the 3D point cloud of the bridge from the multi-view images and their masks acquired from the UAV. The 3D reconstruction technique based on stereo vision principles is also a research hotspot and a technology with wide-ranging applications in the field of computer vision. Therefore, numerous algorithms and complete methods have been proposed for each step of the reconstruction process, and open-source as well as commercial 3D reconstruction software based on these methods and algorithms have been widely proposed and utilized. Among them, the open-source software COLMAP is a versatile framework integrating SFM and multi-view stereo (MVS) [32, 33]. It features both graphical and command-line interfaces and provides interfaces for inputting user-detected feature results and selecting various commonly used algorithms, thus possessing good potential for secondary development. Therefore, COLMAP is adopted as the tool for 3D reconstruction in this section.



(a) The precision curve of the training process (b) The P-R curve of the network

Figure 6. Training of bridge segmentation network.

During the process of using COLMAP for 3D reconstruction, it consists of two parts: image preprocessing and reconstruction. Preprocessing involves detecting features in the images and matching them, while reconstruction includes sparse point cloud reconstruction, bundle adjustment, and dense point cloud reconstruction. In this section's reconstruction, since the camera used in this study is a rigorously calibrated mapping camera, the camera model chosen for feature detection in the preprocessing part is the pinhole camera model, and the feature detection method used is the SIFT feature detection provided by COLMAP itself. The images captured by the UAV during flight are sequential image sequences, so Sequential Matching is used for feature matching and geometric consistency verification. After completing the first step, the next step is to start incremental sparse reconstruction. In this step of sparse reconstruction, the program adopts the method proposed by Schönberger and Frahm [32], where the algorithm registers new images on selected initial viewpoints and triangulates them to continuously increase the number of sparse points in the scene. After completing sparse point cloud reconstruction, Bundle Adjustment is applied to optimize and adjust the global image poses. Once the sparse point cloud of the reconstructed scene and the pose of each image are obtained, the next step is to use MVS to recover denser geometric shapes of the scene. The method applied in the program is proposed by Schönberger *et al* [33], which is a depth-map-based method. This method generates depth and normal maps for all registered images, fusing depth and normal maps into a dense point cloud with normal information, and ultimately uses Poisson reconstruction to fuse the dense point cloud into a dense surface. After obtaining the reconstructed dense point cloud and Poisson Mesh, further use of open-source software Meshlab maps the texture of the images onto the mesh, obtaining a textured mesh model. For the textured mesh model with additional image texture, the center point of the bridge bottom is determined from the four endpoints of the bridge bottom, and the normal vector of the orthographic projection is obtained. The bridge bottom is then projected based on the projection centre and normal vector, ultimately obtaining a panorama of the bridge bottom.

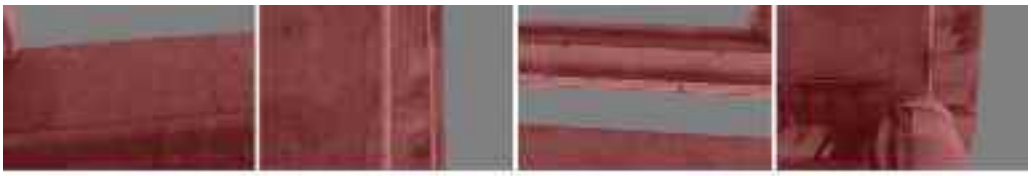
Taking the bottom image of a 50 m-span concrete bridge acquired through close-range photography using a UAV as an example, the above method was employed for processing, resulting in the source image mask, reconstructed sparse point cloud, reconstructed mesh model, and projected panorama image of the bridge bottom as shown in figure 7. For a 50 m-span bridge, the size of the obtained panorama image of the bridge bottom is 30 254 $\times$ 12 044 pixels.

2.3. Damage recognition and localization based on improved object detection network

After obtaining the ultra-high-resolution panorama image of the bridge bottom, the next step is to apply object detection networks to automatically identify defects and calculate their positions in the panorama image based on the bounding boxes of each defect. Unlike the previous bridge targets, the defects in this section are generally smaller in scale, with the pixel area occupied by each defect much smaller than 0.08% of the entire image, posing a typical small object detection problem. Moreover, the input image size in this section can reach hundreds of millions of pixels, making existing networks unable to handle the task. Therefore, this section introduces improvement strategies for existing networks to meet the defect identification requirements.

In section 2.2, the YOLO v8-seg network used was discussed. Considering that the YOLO v8 series networks are state-of-the-art, this section also selects the object detection network from the YOLO v8 network series as the baseline network for defect detection, and further optimizes and improves it to adapt to the specific conditions of this section. The specific improvement strategies are divided into two parts: addressing the issue of oversized images that cannot be input into a network with an input size of only 640 $\times$ 640 pixels by adding a slicing-aided inference module, and modifying the internal structure of the network to enhance small object detection effectiveness.

2.3.1. Adding the slice-assisted inference module. For processing oversized images, an effective approach is to cut them



(a) Image of segmented bridge components

(b) Sparse and dense point clouds

(c) Panoramic of the bridge bottom

Figure 7. Obtaining the panoramic of a 50 m span bridge using the proposed method.

into smaller images according to certain rules and then process these smaller images. This method can avoid the problem of texture loss caused by directly downsizing the image. However, when cutting the images, the integrity of the defects needs to be considered to avoid cutting a single defect into multiple parts. The slicing-aided hyper inference module is a universal framework for detecting large-scale images through image slicing to ensure the accuracy of small object detection, and it can be applied to almost any existing object detection network [34]. This module divides the image into several regions using a sliding window, predicts each region separately, and also performs inference on the entire image. Then, the predicted results of each region are merged with the predicted results of the entire image, and finally filtered using non-maximum suppression. This module can be applied in both fine-tuning and inference phases. During inference, the image is divided into many smaller sub-images, resized, and then fed into the model for prediction. The predicted results are then transformed back to the original image coordinates after non-maximum suppression. The principle of this module is illustrated in figure 8.

In this section, the SAHI module is applied during the inference phase. During the application process, the size of the sliding window and the overlap between adjacent sliding windows will impact the precision of model inference. An excessively large sliding window size can result in the loss of texture when resizing sub-images to fit the model, while the overlap rate may inadvertently segment a single anomaly into two when an anomaly exists at the boundary of sub-images. The network utilized in this section is an optimization based on the YOLO v8 network. Consequently, the network's input size is consistent with YOLO v8, set at 640 $\times$ 640 pixels. Considering that a small sliding window size significantly increases the computational time for sliding recognition, and a large size results in the loss of details, this section sets the sliding window size to be consistent with the image size in the dataset created for this study (800 $\times$ 800 pixels). To determine the overlap rate, a statistical analysis of anomaly sizes in 1000 collected images containing anomalies was conducted. The results indicate that the median length of annotated boxes is 71 pixels, with 90% of boxes having a length not exceeding 160 pixels, and 95% of boxes having an area not exceeding 8% of the image area. Therefore, this section adopts a sliding window overlap rate of 0.2, calculated as the ratio of 90% of the annotated box length to the image length.

2.3.2. Modification strategy of network structure. The objective of model optimization in this section is to improve the recognition performance of small-sized damages, focusing on enhancing the detection of small targets. YOLO v8 networks includes an improved version for detecting small objects, namely YOLO v8-p2, which incorporates an additional image pyramid to extract multi-scale features, enabling

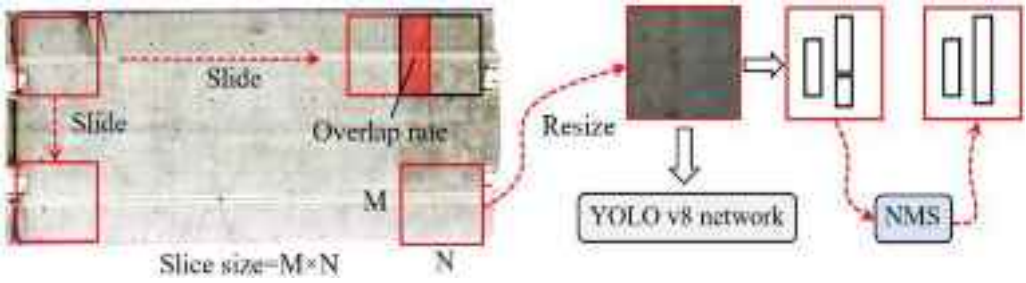


Figure 8. The principle of slicing aided hyper inference module.

more accurate detection and localization of smaller objects compared to the original YOLO v8 model. However, when training and testing this network on the dataset established in this study, the achieved final accuracy still falls short. Therefore, in addition to this network, further optimization strategies are adopted in this section by introducing the plug-and-play omni-dimensional dynamic convolution (ODConv) module, incorporating a multi-dimensional attention mechanism to learn more flexible attention over the four dimensions of convolutional kernel space [35].

ODConv is a dynamic convolution module where the convolution operation, being one of the core components of convolutional neural networks, usually involves multiplying the input by a static convolutional layer to obtain the output. In dynamic convolution, however, instead of a single convolutional kernel, it employs n convolutional kernels W_i, each multiplied by a weight $\alpha_{\omega i}$ computed based on the input features and then summed up, yielding the final output feature y, as expressed by the formula $y = (\alpha_{\omega 1} W_1 + \ldots + \alpha_{\omega n} W_n) \times x$. Recent studies on convolutional neural networks have shown that dynamic convolution allows the network to learn linear combinations of multiple convolutional kernels and weight them based on the input-related attention, significantly enhancing the accuracy of the network while maintaining efficient inference. Therefore, replacing the original convolution module with a dynamic convolution module on mainstream networks is considered an effective strategy to improve network accuracy [36]. For instance, Zhang *et al* applied the dynamic convolutional network DyNet on networks like ShuffleNetV2, resulting in a 37.0% reduction in computational cost and a 2.9% increase in accuracy [37]. ODConv is a recently proposed dynamic convolutional module that can be seen as an extension of CondConv. It expands the dynamic properties of CondConv from one dimension and considers the dynamic nature across spatial, input channel, and output channel dimensions, thus representing a full-dimensional dynamic convolution. As evident from the feature output calculation formula above, the most critical aspect of dynamic convolution lies in computing the weights using attention mechanisms. Therefore, ODConv focuses on optimizing attention mechanisms. On one hand, unlike CondConv and other dynamic convolutions, which employ an improved SE attention structure for calculating the weights $\alpha_{\omega i}$, ODConv differs in that CondConv uses the Sigmoid function for calculation. On the other hand, in the overall weight calculation, ODConv introduces three new attention mechanisms: convolutional kernel spatial dimension α_{si}, input channel dimension α_{ci}, and output channel dimension α_{fi}. Consequently, the calculation of ODConv's feature output is modified as follows:

$$y = (\alpha_{\omega 1} \odot \alpha_{fi} \odot \alpha_{ci} \odot \alpha_{si} W_1 + \ldots + \alpha_{\omega n} \odot \alpha_{fn} \odot \alpha_{cn} \odot \alpha_{sn} W_n) \times x. \quad (2)$$

The input x is incrementally computed by multiplication with convolutional kernels along spatial positional dimensions, channel dimensions, filter dimensions, and convolutional kernel dimensions. This process allows the input to gather rich contextual information. The network architecture after incorporating the slicing aided hyper inference module and ODConv module into the YOLO v8-p2 network is illustrated in figure 9.

2.3.3. Establishment of the damage dataset. After determining the network structure for this section, it is necessary to create a dataset for training the improved network specifically for detecting exposed reinforcement bars and corrosion on concrete bridges. The images in the dataset established in this study are derived from a large number of images obtained from UAVs during the inspection of in-service concrete bridges in the preliminary stage. From these images, 2000 images were selected as the original images for the dataset, and their sizes were uniformly adjusted to 800 × 800 pixels through standard image segmentation. The dataset was augmented using the same data augmentation methods as described in section 2.2, resulting in a final dataset of 8000 images containing defects. Since defect detection in this section falls under the category of object detection tasks, the manual labeling process was conducted using the open-source labeling tool. An example of images in the defect database is shown in figure 10.

To analyze the improvement in defect detection performance of the modified network compared to the original YOLO v8 network, three different networks were trained on the established dataset: YOLO v8x, which has the largest size and highest accuracy in the YOLO v8 network, YOLO v8-p2, which is optimized for small targets, and the improved network. The training framework and hardware used were consistent with those described in section 2.2, and the network

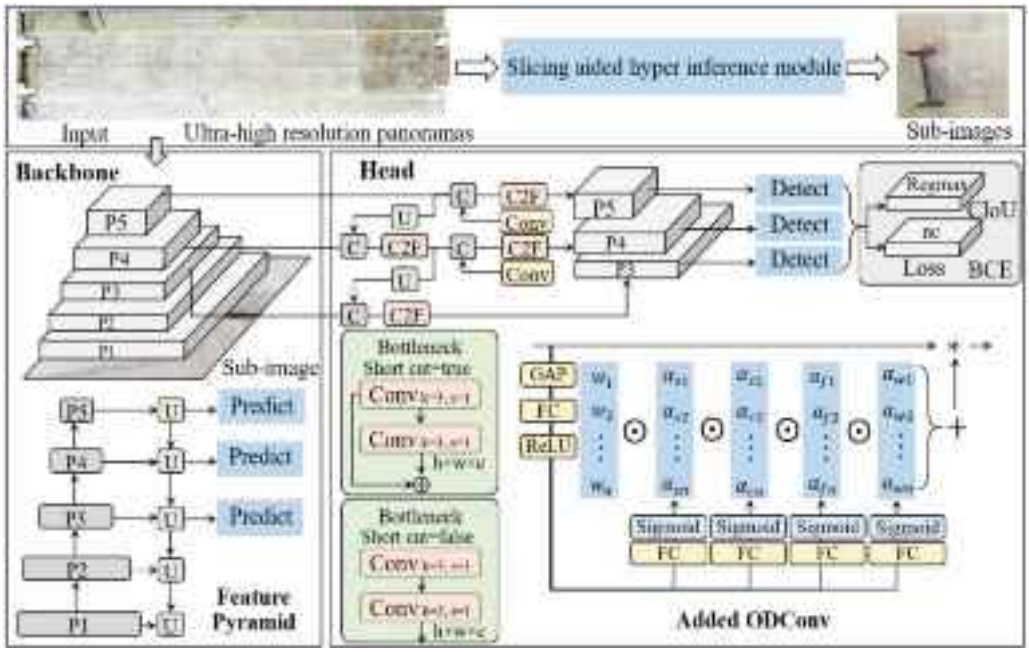
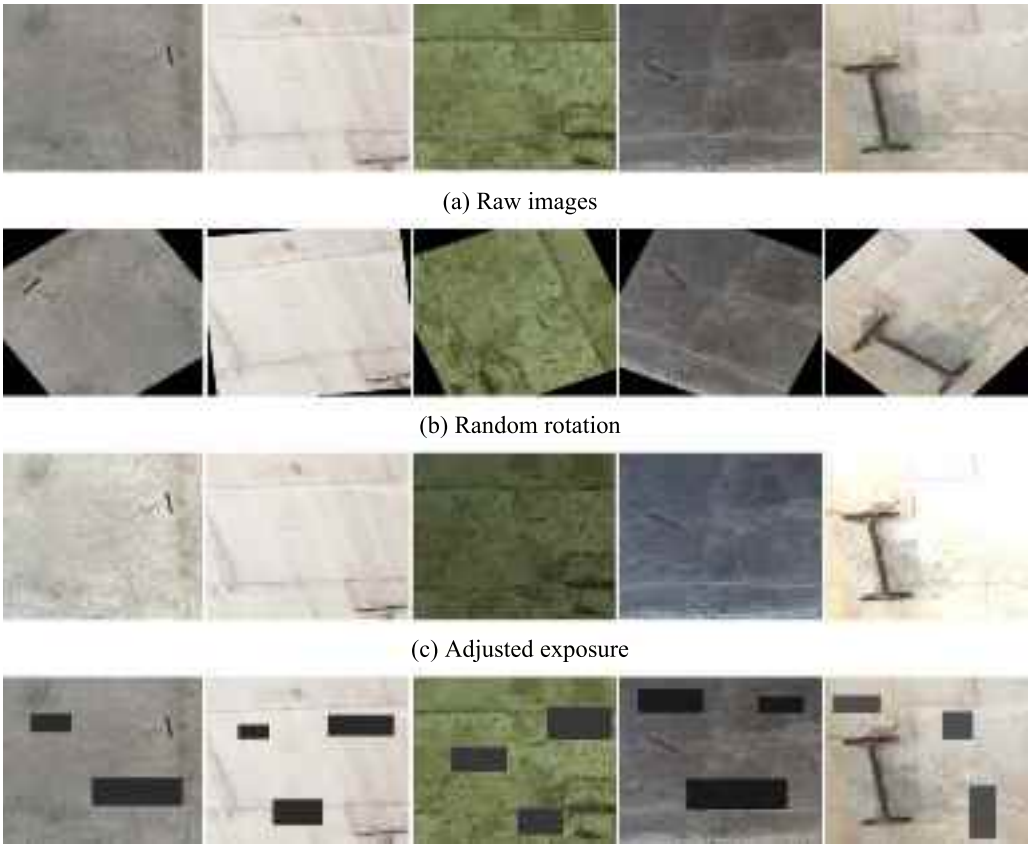


Figure 9. The structure of the improved network.



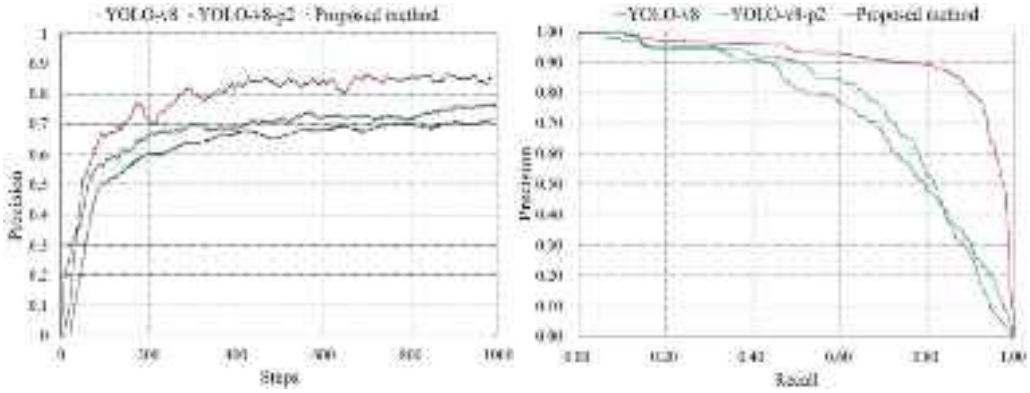
(a) Raw images

(b) Random rotation

(c) Adjusted exposure

(d) Random mesh occlusion

Figure 10. The images in the established damage dataset.



(a) The precision curve of the training process (b) The P-R curve of the networks

Figure 11. The training curves of three networks during the training process.

(a) Overall view of the bridge (b) Exposed steel bars and their repair

Figure 12. Overview of the bridge and types of major damage.

parameters during training were kept the same as those of the original networks. The training was conducted for 1000 steps, and the parameter changes of the three networks during training are compared in figure 11. The training results indicate that the precision of the original YOLO v8x network is 0.716, while the precision of the YOLO v8-p2 network is 0.764. In contrast, the precision of the improved network reaches 0.862, representing a 20.4% improvement over the original network and a 12.8% improvement over the p2-type network.

After obtaining the trained network, it is capable of damage recognition and localization on the panoramic image of the bridge bottom obtained in the previous reconstruction step. Using the trained network for damage recognition on the entire panoramic image involves drawing bounding boxes around the positions where damages are present, and damage localization involves positioning these bounding boxes. Since the panoramic image is obtained through 3D reconstruction, the image possesses scaling parameters after projection. Therefore, the localization of bounding boxes requires obtaining their coordinates in pixel units. In this study, a coordinate system is established with the left side of the bridge bottom as the *y*-axis and the centerline of the bridge bottom as the *x*-axis. The center coordinates of each bounding box are determined in this coordinate system and, when multiplied by the scaling parameters, provide the true localization results of damages in this coordinate system.

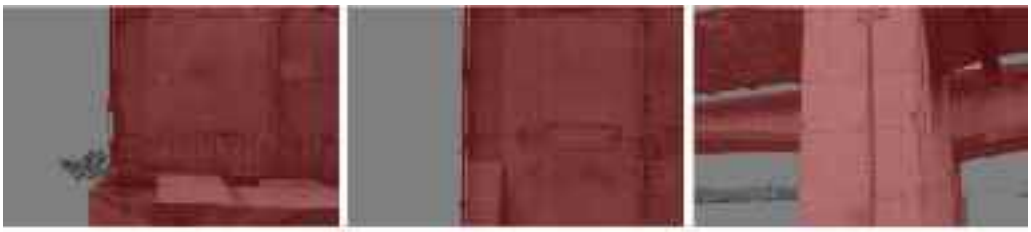
3. Field test on an in-service bridge

3.1. Bridge overview and data acquisition

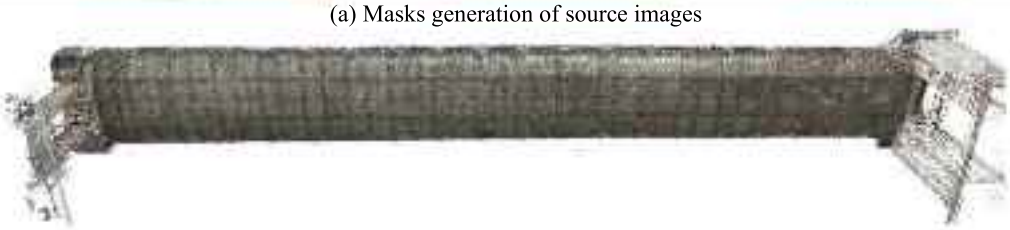
The bridge tested in this study is a prestressed concrete bridge with a single-tower cable-stayed structure, which was put into service in 1988. It has a total length of 1675.2 m, with the main bridge consisting of two spans of 160 m each and a deck width of 16 m. Due to its location in a highly trafficked area, the bridge experiences a daily traffic volume of up to 20 000 vehicles. After more than thirty years of service, the bridge has developed a certain number of surface defects. According to previous manual inspections, the most significant defects are steel bar exposure and corrosion caused by concrete damage. Therefore, repair work has been carried out on areas with exposed steel bars and corrosion. This inspection serves both as a routine examination of the bridge's defects and as an assessment of the number of new instances of steel bar exposure and corrosion since the last repair, in order to evaluate the timing of the next repair. The overall view of the bridge and the condition of the defects are illustrated in figure 12.

The testing in this study focused on the bottom surface of the 160 m main span on the west side of the bridge. During the data acquisition process, an UAV was employed following the data capture method outlined in section 2.1. The UAV captured images by flying vertically upwards in four parallel flight lines beneath the bridge, as well as capturing oblique

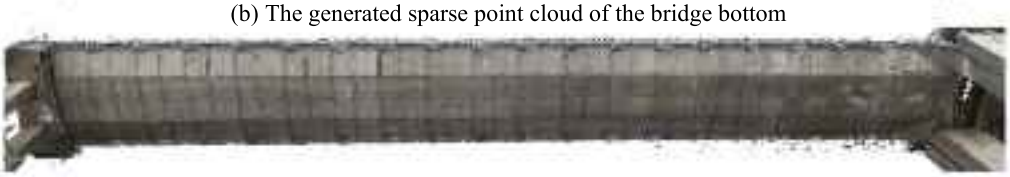
Figure 13. Image taken upwards (a) and image taken tilted 45 ° from both sides (b, c).



(a) Masks generation of source images



(b) The generated sparse point cloud of the bridge bottom



(c) The generated dense point cloud of the bridge bottom

(d) Panoramic view of the bridge bottom

Figure 14. 3D reconstruction and panoramic view of the bridge bottom.

images from the sides at a 45° angle. The entire data acquisition process took approximately one hour, resulting in a total of 2494 images with a resolution of 8192 × 5460 pixels. The images captured from the three different angles are illustrated in figure 13. Throughout the data acquisition, the camera was set with fixed exposure time and sensitivity. Due to variations in sunlight conditions, slight differences in brightness may exist among the acquired images.

3.2. Bridge bottom 3D reconstruction and panoramic image generation

The 2494 acquired images underwent 3D reconstruction and panoramic image generation using the methodology outlined in section 2.2. Initially, the trained YOLO v8-seg network was employed to generate masks for the source images. The resulting mask is depicted in figure 14(a). Subsequently, both the source images and masks were input into the 3D reconstruction algorithm, involving sequential steps of feature detection, feature matching, sparse point cloud reconstruction, and dense point cloud reconstruction. The reconstructed results of the bridge underside are illustrated in figures 14(b) and (c). The sparse point cloud comprises 403 000 points, while the dense point cloud encompasses 641 million points. Post the completion of point cloud reconstruction using COLMAP, Meshlab was employed for further reconstruction to obtain a textured model of the bridge underside. The projection was performed from a viewpoint directly beneath the bridge, resulting in the panoramic image of the bridge underside with a resolution of 95 451 × 21 805 pixels. After the reconstruction, due to the non-utilization of every pixel in the reconstruction and texture projection processes, the final resolution of the panoramic image was reduced to 1.03 mm/pixel, as depicted in figure 14(d).

(a) Damage labelled using proposed method

(b) Damage labelled manually

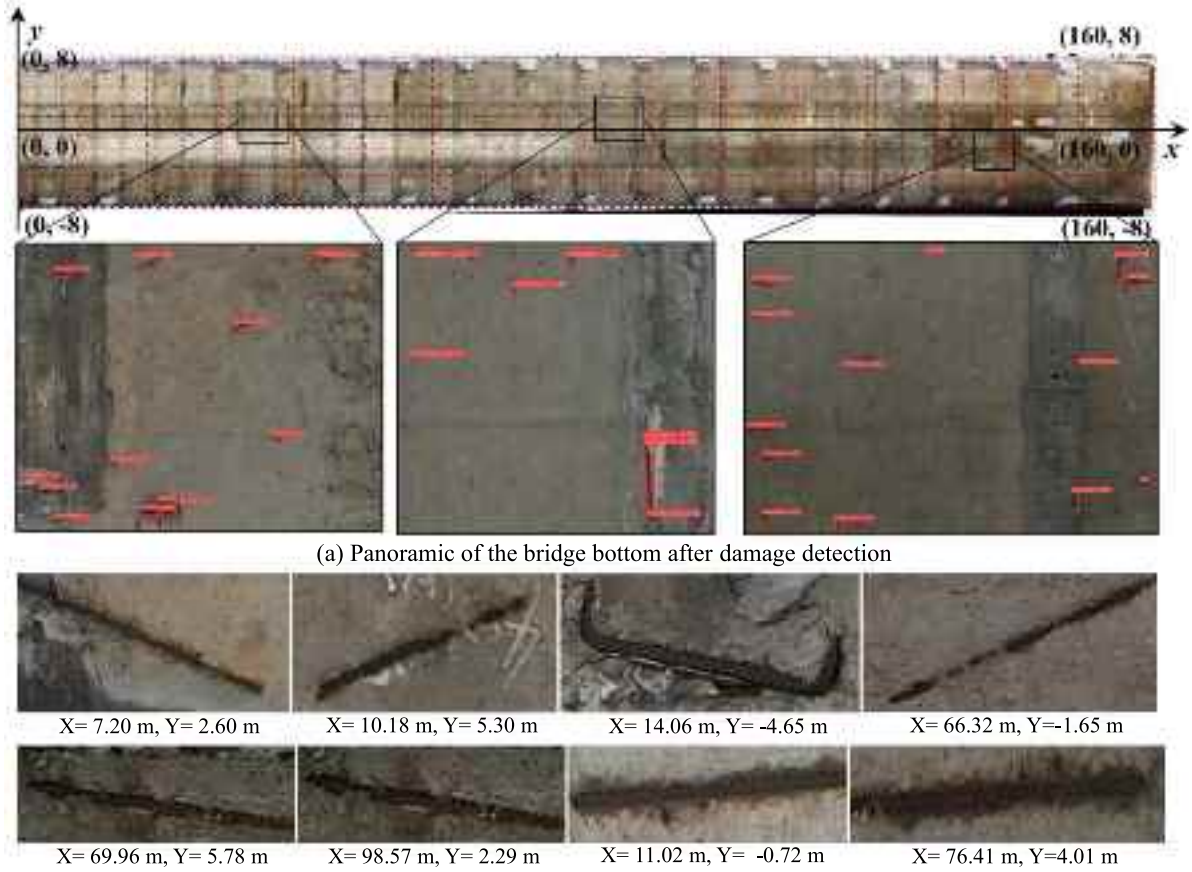
Figure 15. Damage labelled manually and using proposed method.

3.3. Identification and localization of bridge bottom damage

For damage identification, the damage detection network trained in section 2.3 was employed to process the panoramic image. The panoramic image has a length of 95 451 pixels. With a sliding window size of 800 × 800 pixels and an overlap rate of 0.2, the entire sliding process took 318 s. When applying the trained network for the recognition of each sub-image, the time required for a single recognition was 18 m s^{-1}. Therefore, the overall time for damage identification across the entire panoramic image was 383 s.

Regarding damage identification results in the panoramic image under the bridge, a comparison was conducted between manual identification methods and the proposed automated identification method to assess its accuracy. In this section, sub-images obtained during the sliding window process were saved. From these, 50 images containing damages were selected, and manual identification was performed by outlining bounding boxes around the damages. Simultaneously, the proposed method's identified bounding boxes were outputted. A comparison was then made between the manually marked structures and those identified by the proposed method to determine the recognition accuracy (that is whether damage in the images were correctly identified) and bounding box coverage accuracy (that is the overlap ratio of bounding boxes). The results indicated that the proposed method achieved a recognition accuracy of 0.917 and a bounding box coverage accuracy of 0.806. Images marked manually and by the proposed method are shown in figure 15.

After identifying damages in the panoramic image under the bridge in the previous step, the coordinate system for the bridge bottom was established using the method described in section 2.3. The coordinates of the damage bounding boxes on the bridge bottom were outputted, and the average of the coordinates of the top-left and bottom-right corners of each bounding box was taken as the damage location. This process



(a) Panoramic of the bridge bottom after damage detection

(b) Cropped damage images with location information

Figure 16. Localization results of damages at the bottom of the bridge.

yielded the position information for all damages on the bridge bottom. Each identified damage was individually segmented, and the damage location information was assigned. The results are presented in figure 16(b). By mapping all damage information onto the panoramic image of the bridge bottom, a comprehensive and intuitive damage detection result for the entire bridge bottom was obtained, as shown in figure 16(a). To quantitatively analyze the accuracy of the proposed method for the localization of the damages, the obtained panoramic image of the underside of the bridge is identified and labeled with damages manually, and the coordinates of the center of each manually labeled box are output and compared with the coordinates produced by the proposed method. Since the number of damages in the panoramic image is too large and the image size is too wide for manual labeling, the panoramic image is segmented into 28 images with 3000 $\times$ 3000 pixels, and the damages in the images are labeled and counted separately. The maximum and average error of the localization are calculated, and the results are shown in table 1. Compared with manual labeling, the maximum error of localization of the proposed method is 38.89 mm in the x-axis direction and 39.01 mm in the y-axis direction, and the average error is 6.67 mm in the x-axis direction and 6.60 mm in the y-axis direction. For a bridge with a span of 160 m, the localization accuracy of the damage at the centimeter level is satisfactory, proving that the accuracy of the proposed method is sufficient to meet the requirements.

4. Conclusion

This study proposes a method for the 3D reconstruction and damage localization of bridge bottoms based on close-range aerial photography. The approach utilizes images captured by UAV in close proximity to the bridge bottom as input data. A pre-segmentation-based 3D reconstruction algorithm is employed to reconstruct the 3D model of the bridge bottom, and its projection yields an ultra-high-resolution panoramic image of the bridge bottom. An optimized object detection network is then applied to identify damages and establish coordinates for precise damage localization. This methodology enables efficient and fine-grained identification and localization of extensive damages on the bridge bottom. Specific conclusions include:

(1) For locations such as the underside of a bridge, which are challenging to access for routine inspections, employing close-range aerial photography provides the means to obtain sub-millimeter-level image data of the bridge

Table 1. Damages localization errors of the proposed method.

Block	Max-error *x*-coord (mm)	Max-error *y*-coord (mm)	Avg-error *x*-coord (mm)	Avg-error *y*-coord (mm)
1	38.13	34.41	9.47	8.33
2	27.15	31.87	9.53	8.61
3	23.17	26.07	9.04	6.70
4	13.66	15.21	4.87	5.13
5	21.76	19.10	5.77	5.24
6	25.64	27.28	5.14	4.74
7	34.74	29.26	8.08	9.97
8	24.52	20.19	6.96	5.58
9	24.89	24.35	5.87	5.33
10	18.45	17.41	6.77	5.65
11	27.97	35.02	8.80	7.66
12	23.26	36.01	7.04	7.45
13	23.49	26.81	5.59	6.28
14	31.27	39.01	8.27	9.01
15	23.67	26.94	5.13	6.95
16	24.02	21.18	7.42	7.97
17	38.89	32.30	8.64	8.24
18	19.19	15.79	4.20	4.44
19	29.00	32.40	7.05	8.26
20	23.35	26.84	5.68	5.35
21	24.24	17.71	4.70	3.95
22	24.70	32.02	7.99	8.64
23	23.55	14.53	5.17	4.90
24	12.59	18.80	4.02	4.80
25	30.92	19.25	8.86	7.09
26	16.55	20.02	4.17	5.22
27	26.12	21.12	6.08	6.26
28	27.84	28.19	6.45	7.04
Errors	38.89	39.01	6.67	6.60

bottom. Through the proposed method of capturing multiple perspectives with overlapping coverage, the resulting multi-angle images facilitate the 3D reconstruction of the bridge underside. Notably, the authentic resolution of the images can reach 0.63 mm/pixel.

(2) In the context of UAV based inspections, the obtained bridge images often feature complex backgrounds in their surroundings. To address this, the state-of-the-art YOLO v8-seg network is employed for training on a bridge dataset, achieving an impressive bridge segmentation accuracy of 99.5%. The resulting image masks derived from segmentation results are utilized for the 3D reconstruction of the bridge. This approach not only reduces computational overhead but also yields a clean and pure 3D model of the bridge. Leveraging the reconstructed model, when projected from the underside of the bridge, enables the generation of a super-high-resolution panorama image of the bridge bottom.

(3) The proposed YOLO v8 network, enhanced with the slicing aided hyper inference module and structural improvements, overcomes challenges encountered by conventional networks in handling ultra-large image sizes and addressing the issue of losing image details that make it challenging to identify small objects. In the established database comprising 8000 images of exposed concrete bridge reinforcements and corrosion, the improved network demonstrates a 20.4% increase in accuracy for small target recognition compared to the original network.

(4) The proposed method was applied to the bottom damage detection of a concrete bridge with a main span of 160 m. Test results demonstrate that the proposed approach can acquire high-resolution multi-perspective images of the bridge bottom, reconstruct a 3D model of the bridge bottom, project a high-resolution panoramic image of the bottom, and automatically identify and output the coordinates of exposed steel reinforcements and corrosion.

The proposed method holds promise to assist or even replace existing manual inspection methods in the routine evaluation of bridges, significantly reducing dependence on manual efforts during the analysis of inspection data. Hence, it demonstrates apparent practicality and application prospects. However, the present study still has limitations that need to be overcome in future work. First, the resolution of the panorama of the bridge obtained in the test conditions is 1.03 mm/pixel, which is sufficient for damage such as concrete spalling and exposed reinforcement corrosion, but for tiny damage, such as fine cracks, they cannot be recognized from the panorama. Meanwhile, calculations of geometric information (such as width and area) of the damages based on this resolution are only available at millimeter level of accuracy. In future studies, images of bridge surfaces will be taken using cameras with longer focal lengths to improve the resolution of panoramic images. The use of algorithms based on sub-pixel edge detection or super-resolution image enlargement is also expected to improve the accuracy of calculating the geometric information of the damage. The control of UAVs in the narrow environment beneath the bridge still relies on manual operation, preventing the achievement of full automation in data collection. Future research efforts will focus on developing automated methods for UAVs to follow predefined flight paths, aiming to further enhance the automation level of the proposed approach. Additionally, due to the time-consuming nature of the bridge bottom's 3D reconstruction process, the data analysis time of this method remains relatively long. Future investigations will explore deep learning-based 3D reconstruction methods to overcome the time-consuming aspects of traditional feature detection and matching-based 3D reconstruction algorithms.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Funding

This research is supported by the Project of Industry Foresight and Key Core Technologies (Grant No. BE2021021), the Special Project on Transformation of Scientific and Technological Achievements in Jiangsu Province (No. BA2022009) for which the authors are grateful.

Conflict of interest

The authors declare no conflicts of interest.

ORCID iD

Shang Jiang https://orcid.org/0000-0002-0665-8693

References

[1] Zhang C, Zou Y, Wang F, Del Rey Castillo E, Dimyadi J and Chen L 2022 Towards fully automated unmanned aerial vehicle-enabled bridge inspection: where are we at? *Constr. Build. Mater.* **347** 128543
[2] Li G, Liu Q, Zhao S, Qiao W and Ren X 2020 Automatic crack recognition for concrete bridges using a fully convolutional neural network and naive Bayes data fusion based on a visual detection system *Meas. Sci. Technol.* **31** 075403
[3] Wu Y, Meng F, Qin Y, Qian Y, Xu F and Jia L 2023 UAV imagery based potential safety hazard evaluation for high-speed railroad using Real-time instance segmentation *Adv. Eng. Inf.* **55** 101819
[4] Liu B, Yang T, Wu X, Wang B, Zhang H and Wu Y 2024 UAV imagery-based railroad station building inspection using hybrid learning architecture *Meas. Sci. Technol.* **35** 086206
[5] Jiang S, Gu S and Yan Z 2022 Pavement crack measurement based on aerial 3D reconstruction and learning-based segmentation method *Meas. Sci. Technol.* **34** 015801
[6] Ni F, He Z, Jiang S, Wang W and Zhang J 2022 A generative adversarial learning strategy for enhanced lightweight crack delineation networks *Adv. Eng. Inf.* **52** 101575
[7] Zhang A, Wang K C, Fei Y, Liu Y, Tao S, Chen C and Li B 2018 Deep learning-based fully automated pavement crack detection on 3D asphalt surfaces with an improved CrackNet *J. Comput. Civ. Eng.* **32** 04018041
[8] Liu Y H, Zheng Y Q, Shao Z F, Wei T, Cui T C and Xu R 2024 Defect detection of the surface of wind turbine blades combining attention mechanism *Adv. Eng. Inf.* **59** 102292
[9] Abdallah A M, Atadero R A and Ozbek M E 2022 A state-of-the-art review of bridge inspection planning: current situation and future needs *J. Bridge Eng.* **27** 03121001
[10] Jiang S, Cheng Y and Zhang J 2023 Vision-guided unmanned aerial system for rapid multiple-type damage detection and localization *Struct. Health Monit.* **22** 319–37
[11] Miller I D *et al* 2020 Mine tunnel exploration using multiple quadrupedal robots *IEEE Robot. Autom. Lett.* **5** 2840–7
[12] Kang D and Cha Y J 2018 Autonomous UAVs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging *Comput.-Aided Civ. Infrastruct. Eng.* **33** 885–902
[13] Jiang S, Wu Y and Zhang J 2023 Bridge coating inspection based on two-stage automatic method and collision-tolerant unmanned aerial system *Autom. Constr.* **146** 104685
[14] Wang Y, Zhang X, Zhang M, Sun L and Li M 2021 Self-compliant track-type wall-climbing robot for variable curvature facade *IEEE Access* **10** 51951–63
[15] Mahmood S K, Bakhy S H and Tawfik M A 2021 Propeller-type wall-climbing robots: a review *IOP Conf. Ser.: Mater. Sci. Eng.* **1094** 012106
[16] Yang L, Li B, Feng J, Yang G, Chang Y, Jiang B and Xiao J 2023 Automated wall-climbing robot for concrete construction inspection *J. Field Robot.* **40** 110–29
[17] Nguyen S T, Pham A Q, Motley C and La H M 2020 A practical climbing robot for steel bridge inspection *2020 IEEE Int. Conf. on Robotics and Automation (ICRA)* (IEEE) pp 9322–8
[18] Li R, Yu J, Li F, Yang R, Wang Y and Peng Z 2023 Automatic bridge crack detection using Unmanned aerial vehicle and Faster R-CNN *Constr. Build. Mater.* **362** 129659
[19] Ni F, Zhang J and Chen Z 2019 Pixel-level crack delineation in images with convolutional feature fusion *Struct. Control Health Monit.* **26** e2286
[20] Xu Y, Bao Y, Chen J, Zuo W and Li H 2019 Surface fatigue crack identification in steel box girder of bridges by a deep fusion convolutional neural network based on consumer-grade camera images *Struct. Health Monit.* **18** 653–74
[21] Yoon S, Gwon G H, Lee J H and Jung H J 2021 Three-dimensional image coordinate-based missing region of interest area detection and damage localization for bridge visual inspection using unmanned aerial vehicles *Struct. Health Monit.* **20** 1462–75
[22] Jiang S and Zhang J 2020 Real-time crack assessment using deep neural networks with wall-climbing unmanned aerial system *Comput.-Aided Civ. Infrastruct. Eng.* **35** 549–64
[23] Attard L, Debono C J, Valentino G and Di Castro M 2018 Tunnel inspection using photogrammetric techniques and image processing: a review *ISPRS J. Photogramm. Remote Sens.* **144** 180–8
[24] Zhang Z 2000 A flexible new technique for camera calibration *IEEE Trans. Pattern Anal. Mach. Intell.* **22** 1330–4
[25] Long J, Shelhamer E and Darrell T 2015 Fully convolutional networks for semantic segmentation *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 3431–40
[26] Liu W, Rabinovich A and Berg A C 2015 Parsenet: looking wider to see better (arXiv:1506.04579)
[27] Siddique N, Paheding S, Elkin C P and Devabhaktuni V 2021 U-net and its variants for medical image segmentation: a review of theory and applications *IEEE Access* **9** 82031–57
[28] He K, Gkioxari G, Dollár P and Girshick R 2017 Mask r-cnn *Proc. IEEE Int. Conf. on Computer Vision* pp 2961–9
[29] Fan D P, Ji G P, Zhou T, Chen G, Fu H, Shen J and Shao L 2020 Pranet: parallel reverse attention network for polyp segmentation *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention* (Springer) pp 263–73
[30] Jiang P, Ergu D, Liu F, Cai Y and Ma B 2022 A review of Yolo algorithm developments *Proc. Comput. Sci.* **199** 1066–73
[31] Terven J, Córdova-Esparza D M and Romero-González J A 2023 A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS *Mach. Learn. Knowl. Extraction* **5** 1680–716
[32] Schönberger J L and Frahm J M 2016 Structure-from-motion revisited *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 4104–13

[33] Schönberger J L 2018 Robust methods for accurate and efficient 3D modeling from unstructured imagery *Doctoral Dissertation* ETH Zurich
[34] Akyon F C, Altinuc S O and Temizel A 2022 Slicing aided hyper inference and fine-tuning for small object detection *2022 IEEE Int. Conf. on Image Processing (ICIP)* (IEEE) pp 966–70
[35] Li C, Zhou A and Yao A 2022 Omni-dimensional dynamic convolution (arXiv:2209.07947)
[36] Yang B, Bender G, Le Q V and Ngiam J 2019 Condconv: conditionally parameterized convolutions for efficient inference (arXiv:1904.04971v3)
[37] Zhang Y, Zhang J, Wang Q and Zhong Z 2020 Dynet: dynamic convolution for accelerating convolutional neural networks (arXiv:2004.10694)