Research Article

Computer Vision-Based Detection for Delayed Fracture of Bolts in Steel Bridges

Jing Zhou and Linsheng Huo

State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian 116024, China

Correspondence should be addressed to Linsheng Huo; lishuo@dlut.edu.cn

Received 3 July 2021; Revised 8 October 2021; Accepted 23 October 2021; Published 9 November 2021

Academic Editor: Hongjin Wang

Copyright © 2021 Jing Zhou and Linsheng Huo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The delayed fracture of high-strength bolts occurs frequently in the bolt connections of long-span steel bridges. This phenomenon can threaten the safety of structures and even lead to serious accidents in certain cases. However, the manual inspection commonly used in engineering to detect the fractured bolts is time-consuming and inconvenient. Therefore, a computer vision-based inspection approach is proposed in this paper to rapidly and automatically detect the fractured bolts. The proposed approach is realized by a convolutional neural network- (CNN-) based deep learning algorithm, the third version of You Only Look Once (YOLOv3). A challenge for the detector training using YOLOv3 is that only limited amounts of images of the fractured bolts are available in practice. To address this challenge, five data augmentation methods are introduced to produce more labeled images, including brightness transformation, Gaussian blur, flipping, perspective transformation, and scaling. Six YOLOv3 neural networks are trained using six different augmented training sets, and then, the performance of each detector is tested on the same testing set to compare the effectiveness of different augmentation methods. The highest average precision (AP) of the trained detectors is 89.14% when the intersection over union (IOU) threshold is set to 0.5. The practicality and robustness of the proposed method are further demonstrated on images that were never used in the training and testing of the detector. The results demonstrate that the proposed method can quickly and automatically detect the delayed fracture of high-strength bolts.

1. Introduction

High-strength bolt connections are widely used to assemble the load-bearing members of the steel structure in long-span steel bridges. The popularity of the bolt connections is attributed to their low cost, high reliability, and rapid assembly [1]. However, these bridges are often operated in adverse environments and subject to corrosion [2, 3], vibration and fatigue [4, 5], and thermal cycling, which can contribute to the damage of bolts. The damage types of bolts that occur the most include the looseness and delayed fracture. The delayed fracture of bolts refers to the sudden fracture of bolts under constant tension [6]. Due to the huge energy released by the brittle fracture, the fractured bolts will be missing. The damage of bolts will threaten the safety of the bridges and may even lead to severe accidents. Hence, it is necessary to monitor the bolt condition in the daily operation and maintenance phase.

Over the decades, structural health monitoring methods have attracted lots of attention [7–10], and they have been applied to detect the bolt damage [11, 12]. They mainly rely on the sensor technology to identify the variations of the preload, including piezoelectric active sensing methods [13, 14], the electromechanical impedance methods [15, 16], and the vibroacoustic modulation methods [17, 18]. A “smart washer” with a piezoceramic patch sandwiched between two flat metal rings was developed to monitor the bolted joint through the active sensing method [19]. Further, the fluctuation of the impedance signatures in frequency was utilized to evaluate the bolted joint with the developed “smart washer” [20]. A novel vibroacoustic modulation method was proposed to monitor the early looseness of a bolt in real time [21]. Notably, although the contact sensor-based methods are proposed to detect the decrease of preload induced by initial bolt looseness, they can also be used to detect the delayed fracture of bolts,

which is a kind of brittle damage and can result in the disappearance of the preload [22, 23]. Nonetheless, the contact sensor-based methods face the challenge of the cost scaling up when monitoring multiple bolts, because one sensor can only perform the measurement at one bolt. As a result, most bridges are impossible to equip with enough sensors, and the current monitoring method in practice highly relies on manual inspection. However, the whole inspection process is very dangerous and inefficient. As shown in Figure 1, maintenance workers are inspecting the delayed fracture of high-strength bolts on a long-span steel bridge.

In recent years, computer vision technology has received substantial attention as an interdisciplinary subject, and it has been applied in civil infrastructure inspection and monitoring to improve the accuracy and efficiency of manual vision inspection [24, 25]. It has been applied to detect bolt damage because there are always a huge number of bolts in actual steel structures. Park et al. [26] proposed a vision-based method to evaluate the rotation angle of a bolt nut. Cha et al. [27] utilized the image-processing techniques and the support vector machine to detect the bolt looseness. However, traditional computer vision-based methods have poor robustness and low accuracy. On the other hand, the convolutional neural network (CNN) has gained great success in computer vision [28] with the development of deep learning technology, and CNN-based algorithms have achieved the most advanced performance in various tasks, including image classification [29], object detection [30], and semantic segmentation [31]. This kind of technology has also been applied for bolt damage detection. Huynh et al. [32] proposed a quasiautonomous bolt looseness detection method, where the plausible bolts were detected using a CNN-based object detector and the rotation angle of each bolt was measured by the Hough line transform. Zhao et al. [33] proposed a method for the measurement of the bolt-loosening rotation angle using a CNN-based object detector. Wang et al. [34] designed a computer vision-based method by integrating the perspective transformation to detect the bolt looseness for flange connections. However, most studies have only focused on the detection of bolt looseness, and there is no research on the inspection of the delayed fractures in high-strength bolts, to the authors' best knowledge. The visual characteristics of the delayed fracture of high-strength bolts are totally different from looseness, because the fractured bolts will be missing due to the tremendous amount of energy released by the fracture [22, 23]. Notably, bolt delayed fracture can be more dangerous than bolt looseness in theory, because the former will cause the vanishing of the preload, whereas the latter will only reduce the force. Hence, this paper proposed a computer vision-based inspection method for the delayed fracture of bolts, where the damage was detected and located in an automated fashion using an object detection algorithm.

The task of the object detection is to classify and locate the targets in the image, and various algorithms have been developed with high recognition accuracy. The CNN-based object detection methods can be divided into region-based and region-free classifications according to the differences in the idea of the algorithm. The region-based approaches, such as the region-based convolutional neural network (R-CNN) [35], Fast R-CNN [36], and Faster R-CNN [37], combine

FIGURE 1: Maintenance workers are inspecting the bolt delayed fracture in a long-span steel bridge.

region proposals and CNN to detect objects. The region proposals are produced from the input image, and they are treated as the set of candidate detections. The region-free methods, such as Single Shot MultiBox Detector (SSD) [38], You Only Look Once (YOLO) [39], YOLOv2 [40], YOLOv3 [41], and YOLOv4 [42], frame the object detection task as a regression problem, and these methods directly detect objects from the input image by using CNN. The speed of region-based methods is slower than region-free methods due to the necessity of region proposals. Hence, the region-free methods were selected in our research for the real-time detection. In addition, YOLOv3 boasts improved performance for detecting small objects in wide-scale images [43, 44]. The size of the delayed fracture of bolts is relatively small in an image of a bolt connection. Therefore, YOLOv3 is selected to detect the delayed fracture of bolts.

On the other hand, the performance of the CNN-based object detector heavily relies on extracting information from abundant labeled images, and the performance can be improved with the increase of training data in amount and diversity. However, it is quite difficult to acquire enough labeled images in practice, and then, the performance of the trained detector is always limited to some extent. For the bolt damage detection task in long-span steel bridges, images are difficult to be captured due to the environmental complexity and limitation (such as the positions of fractured bolts in a long-span bridge are inaccessible in most cases), and the manual labeling of the images is laborious due to the concentration of bolts.

Data augmentation is one of the most commonly used methods to alleviate this problem, and it can automatically enlarge the dataset by utilizing the existing images [45, 46]. In recent years, many data augmentation methods have been developed for object detection, and images are augmented by many kinds of image-processing technologies. The widely used technologies include brightness transformation, flipping, noise addition, and perspective transformation [47]. For example, Fast R-CNN and Faster R-CNN use horizontal flipping to augment training data [36, 37]. The perspective transformation was introduced to enlarge the training dataset for transmission-line object detection [48]. Although many augmentation techniques are available, the selection of the techniques is task-specific and primarily depends on the experience. Thus, the augmentation effects are still unclear for each method in the

detector training of fractured bolts, and it is necessary to study the effectiveness of different data augmentation methods.

In this paper, a computer vision-based inspection method is developed to automatically detect and locate the bolt delayed fracture, and a series of data augmentation methods are utilized to improve the performance of the detector without external laboriousness. In addition, the impact of different data augmentation methods on the performance of the detector is analyzed.

2. Methodology

2.1. Workflow of the Detection Method for the Bolt Delayed Fracture. As shown in Figure 2, the whole process involves three steps, including dataset preparation, detector training, and damage detection. During dataset preparation, many raw images of high-strength bolt connections are first collected through a camera device. Then, the labeled images can be obtained by artificial labeling and data augmentation, where the damage is labeled by enclosing rectangle bounding boxes. The pairwise images and labels are used to train the YOLOv3 neural network until it can pass the performance checking. Finally, the trained neural network can be used as a damage detector to perform damage detection in the real world.

2.2. Overview of YOLOv3 Detector. YOLOv3 is evolved from its predecessors: YOLO and YOLOv2, which mainly improves the detection accuracy, especially for the detection of small targets. Specifically, a new network, Darknet53, integrating residual networks and Darknet19 (the network used in YOLOv2) was introduced to improve the ability of feature extraction, and the multiscale prediction is used to help simultaneously obtain semantic information and fine-grained information from different feature maps. The architecture of YOLOv3 is shown in Figure 3.

At the beginning of the training process, the image-label pairs are fed into the neural network. Each input image is adjusted to a fixed size, and then, it is divided into $S \times S$ grids using upsampling and feature fusion operations. Each grid is tasked with detecting objects that have their center coordinates enclosed by the grid. Each grid outputs b bounding boxes and c conditional category probability. Each bounding box can be determined by the coordinate information (x , y , w , and h) and the confidence score (S_c). The coordinates (x , y) point towards the center of the bounding box. The parameters w and h are, respectively, the width and height of the bounding box. S_c can be obtained according to Equation (1). The loss function value is calculated using the prediction value and label value. The adjustable parameters in the neural network are updated using a backpropagation algorithm. The process is repeated until the loss function value converges at a small value. During the inference process, only the image is fed into the trained neural network, and the prediction of the neural network is regarded as the detection result.

$$S_c = P(\text{Object}) \times \text{IOU}_{\text{pred}}^{\text{truth}}, \quad P(\text{Object}) \in \{0, 1\}, \quad (1)$$

where $P(\text{Object})$ is equal to 0 when no object exists in the grid; otherwise, its value is 1. $\text{IOU}_{\text{pred}}^{\text{truth}}$ is the intersection over union

(IOU) between the predicted bounding box and the ground truth of the object.

The loss function in YOLOv3 consists of three parts: coordinate loss, IOU loss, and classification loss. All of them correspond to the output of the neural network prediction. However, the classification loss is removed in this paper, because the number of classifications is only one. The loss function used in this paper is shown in the following equation:

$$\begin{aligned} \text{loss} = & \lambda_{\text{coord}} \sum_{i=1}^{S \times S} \sum_{j=1}^B I_{ij}^{\text{obj}} (2 - w_i \times h_i) \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\ & + \lambda_{\text{coord}} \sum_{i=1}^{S \times S} \sum_{j=1}^B I_{ij}^{\text{obj}} (2 - w_i \times h_i) \left[(w_i - \hat{w}_i)^2 + (h_i - \hat{h}_i)^2 \right] \\ & - \sum_{i=1}^{S \times S} \sum_{j=1}^B I_{ij}^{\text{obj}} \left[\hat{S}_{ci} \log(S_{ci}) + (1 - \hat{S}_{ci}) \log(1 - S_{ci}) \right] \\ & - \lambda_{\text{noobj}} \sum_{i=1}^{S \times S} \sum_{j=1}^B I_{ij}^{\text{obj}} \left[\hat{S}_{ci} \log(S_{ci}) + (1 - \hat{S}_{ci}) \log(1 - S_{ci}) \right], \end{aligned} \quad (2)$$

where λ_{noobj} and λ_{coord} are the efficiencies of the IOU loss and coordinate loss, respectively; $\hat{x}_i, \hat{y}_i, \hat{w}_i, \hat{h}_i, \hat{S}_{ci}$ are the ground truth values. The value of I_{ij}^{obj} is 1, when the target falls into the j_{th} bounding box of the i_{th} grid; otherwise, it is equal to 0.

In addition, the average precision (AP) is used as an indicator to estimate the performance of the damage detector. The AP sums up the precision-recall curve by computing the area under the curve [49]. The precision (P) and recall (R) are defined as follows:

$$\begin{aligned} P &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ R &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \end{aligned} \quad (3)$$

where true positives (TP) indicate the number of fractured bolts correctly detected by the detector. False positives (FP) point to the number of other objects in the background falsely detected as fractured bolts. False negatives (FN) refer to the number of fractured bolts missed by the detector.

2.3. Data Augmentation Methods to Improve the Detector Performance. To improve the performance of the detector, five data augmentation methods were introduced in this paper, including brightness transformation (BT), Gaussian blur (GB), flipping (FL), perspective transformation (PT), and scaling (SC). The data augmentation methods are selected considering the practical change of the captured images in engineering and the label preserving ability after augmentation. The BT can mimic images taken under different light intensity conditions. The GB can simulate vague images taken under some unfavorable situations, such as long-distance, slightly out of focus, and foggy weather. The PT can imitate images taken from different viewpoints, including the positions that the camera device cannot reach. The FL can further produce new images captured from different viewpoints. The SC can simulate the image

FIGURE 2: Flowchart of the proposed bolt delayed fracture-detection method.

FIGURE 3: The architecture of YOLOv3 neural network.

resolution changes. The sample images after data augmentation are shown in Figure 4, where the labels of the image are represented by green rectangular bounding boxes.

The data augmentation methods take an image and its label as input and automatically generate a new augmented image and corresponding label. As shown in Figure 4, the BT and GB do not change the coordinates of the bounding

box, whereas the PT, FL, and SC can result in the coordinate change of the bounding box. Hence, the bounding boxes after BT and GB are the same as the original bounding boxes, and the bounding boxes after FL, PT, and SC should be rectified.

The details of the data augmentation methods are described in the following text. The linear BT was used in

FIGURE 4: Sample labeled images with different augmentation methods: (a) original image, (b, c) brightness transformation, (d, e) Gaussian blur, (f, g) flipping, (h, i) perspective transformation, and (j, k) image scaling.

this paper, which directly multiplies the pixel value of the image by a certain coefficient. The GB takes the average value of pixels around a certain point as the pixel value of that point, and the surrounding pixels are assigned different weights according to the distance and the normal distribution. The FL consists of vertical flipping and horizontal flipping, and the rectified bounding box can be obtained easily according to the symmetry. The bilinear interpolation technique was used to change the image resolution, and the rectified image and bounding box can be obtained according to the scaling coefficient. The PT transforms an image from one plane into another plane by a perspective transformation matrix, as shown in Equation (4). The components of the matrix can be obtained according to four pairs of points following Equation (5). As shown in Figure 5, four vertices (A_0 , B_0 , C_0 , and D_0) of the input image and four random

sampling points (A_0^* , B_0^* , C_0^* , and D_0^*) of the augmented image were used to calculate the perspective transformation matrix, and the coordinates of each point can be obtained following Equations (6) and (7). After obtaining the perspective transformation matrix, a rectangular bounding box ($A_1B_1C_1D_1$) in the input image can be transformed to a quadrangle bounding box ($A_1^*B_1^*C_1^*D_1^*$) in the augmented image following Equation (4). However, the nonrectangular bounding boxes cannot be trained by CNN. To tackle this problem, the nonrectangular bounding boxes are rectified using label alignment to generate the new rectangular bounding boxes automatically. To make sure the generated rectangular bounding box totally contain the bolt damage, we let $x_{1PT} = x_{2PT} = \min \{x_1^*, x_2^*, x_3^*, x_4^*\}$, $y_{1PT} = y_{3PT} = \min \{y_1^*, y_2^*, y_3^*, y_4^*\}$, $x_{3PT} = x_{4PT} = \max \{x_1^*, x_2^*, x_3^*, x_4^*\}$, and $y_{2PT} = y_{4PT} = \max \{y_1^*, y_2^*, y_3^*, y_4^*\}$. The

FIGURE 5: Flowchart of the perspective transformation data augmentation.

generated rectangular bounding box can be represented by $A_{1PT} = (x_{1PT}, y_{1PT})$, $B_{1PT} = (x_{2PT}, y_{2PT})$, $C_{1PT} = (x_{3PT}, y_{3PT})$, and $D_{1PT} = (x_{4PT}, y_{4PT})$. The augmented image and the original image have the same size, and the blank area in the augmented image is filled with black pixels.

$$u_i = T v_i = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & 1 \end{bmatrix} v_i, \quad (4)$$

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1 x_1^* & -y_1 y_1^* \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2 x_2^* & -y_2 y_2^* \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3 x_3^* & -y_3 y_3^* \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x_4 x_4^* & -y_4 y_4^* \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1 y_1^* & -y_1 y_1^* \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2 y_2^* & -y_2 y_2^* \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3 y_3^* & -y_3 y_3^* \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -x_4 y_4^* & -y_4 y_4^* \end{bmatrix} \begin{bmatrix} t_{11} \\ t_{12} \\ t_{13} \\ t_{21} \\ t_{22} \\ t_{23} \\ t_{31} \\ t_{32} \end{bmatrix} = \begin{bmatrix} x_1^* \\ x_2^* \\ x_3^* \\ x_4^* \\ y_1^* \\ y_2^* \\ y_3^* \\ y_4^* \end{bmatrix}, \quad (5)$$

where T is the projective transformation matrix, $v_i = (x_i, y_i, 1)$ is a point in the original image, and $u_i = (x_i^*, y_i^*, 1)$ is a point in the PT augmented image.

$$x_{tl}, x_{bl}, x_{br}, x_{tr} \in (0, W \times \lambda), \quad (6)$$

$$y_{tl}, y_{bl}, y_{br}, y_{tr} \in (0, H \times \lambda), \quad (7)$$

where W and H represent the width and height of the image; x_{tl} , x_{bl} , x_{br} , x_{tr} , y_{tl} , y_{bl} , y_{br} , and y_{tr} are the distances from points to the corresponding image boundary; and λ is the intensity parameter for perspective transformation; and the greater the λ , the more obvious the perspective phenomenon.

3. Experimental Verification

3.1. Dataset Preparation. Due to the practical challenges of obtaining large amounts of images depicting the bolt delayed fracture in real bridges, only two images of delayed fractured bolts from an actual suspension bridge were collected in this study. It is impossible to train the YOLOv3 neural network with such a limited number of images; however, these two images can be used to demonstrate the practicability of the proposed method. Thus, training images were generated using two steel plates held together with high-strength bolts. Many images of fractured bolts were collected from the fabricated steel plates to train the neural network.

In this paper, a total of 500 raw images were collected at 3016×3016 -pixel and 3016×4032 -pixel resolutions by a smartphone camera from Xiaomi Mi 6. The distance between the object and the camera is approximately from 0.2 m to 1.5 m. To obtain different lighting intensities of a bolt image in an actual bridge, the images were collected outside during different times of the day (e.g., 9 a.m., 1 p.m., and 5 p.m.). The relationship between the camera's viewing direction and the direction of the sunlight illumination will also influence the brightness of the images. Hence, during the image collection, the conditions of front-lighting, back-lighting, and side-

TABLE 1: The number of images in the datasets.

Dataset	DA _{OR}	DA _{BT}	DA _{GB}	DA _{PT}	DA _{FL}	DA _{SC}	Validation set	Testing set
Number	320	960	960	960	960	960	80	100

FIGURE 6: The result of k -means clustering on the original training set.

lighting are all considered. The direction of the camera viewing was set parallel, antiparallel, and perpendicular to the vector of the sunlight, which can, respectively, simulate front-lighting, back-lighting, and side-lighting. Since the shadow from clouds or the bridge structure will affect the detection accuracy, images were also gathered under scattered tree shade. The apparent shape of the bolt changes based on the viewing angle, and thus, images were taken from multiple viewing angles for the same fractured bolt during image acquisition.

After the image acquisition, the fractured bolts in all 500 images were manually labeled with bounding boxes using the custom code written in Python. And considering the convenience of using the dataset in the future, the dataset is converted to PASCAL VOC format [49]. A “.XML” file including the information of the labeled bounding boxes was generated for each image after successful labeling. The file was then converted into a “.txt” file suitable for the training. The labeled images were then randomly divided into three sets: training set, validation set, and testing set, with 320, 80, and 100 images for each set, respectively. During the labeling process, a total of 439 objects were annotated in the 320 training images. The training set was utilized to train the neural network, and the validation set was used to aid the training and avoid overfitting. After training, the performance of the trained detector was estimated with the testing set. Notably, five extra training sets were generated using five data augmentation methods based on the original training set, and finally, a total of six training sets (DA_{OR}, DA_{BT}, DA_{GB}, DA_{FL}, DA_{PT}, and DA_{SC}) were established in this research and used to train six neural networks. DA_{OR} is the original training set with 320 manually labeled images.

TABLE 2: The average precision (AP) of the detectors using different training sets (%).

Training set	DA _{OR}	DA _{BT}	DA _{GB}	DA _{PT}	DA _{FL}	DA _{SC}
AP _{0.5}	84.62	81.81	81.29	89.14	86.37	82.99
Increment	—	-2.81	-3.33	4.52	1.75	-1.63
AP _{0.6}	70.09	75.16	67.31	83.48	75.93	69.39
Increment	—	5.07	-2.78	13.39	5.84	-0.7
AP _{0.7}	42.11	47.55	38.73	60.56	48.87	46.69
Increment	—	5.44	-3.38	18.45	6.76	4.58

DA_{BT}, DA_{GB}, DA_{FL}, DA_{PT}, and DA_{SC} are the training sets consisting of augmented images generated by the corresponding augmentation method and manually labeled images. The augmented images were produced before training for convenience.

Two BT coefficients were randomly selected from 0.6 to 1.4 and utilized to adjust the brightness for images in DA_{OR}, and as a result, 640 new images were generated. The range of the brightness transformation coefficient was determined based on whether the edge of the target can be identified using naked eyes. The original images in DA_{OR} were also modified using GB to generate 640 additional images. The standard deviation for the Gaussian kernel was randomly selected between 0 and 3.0. The range of standard deviations was set in the same manner mentioned in BT. The images in DA_{OR} were horizontally and vertically flipped, and 640 new flipping images were produced. The scaling coefficient was selected from 0.1 to 1.9, and 640 new augmented images were produced. The PT was applied twice, and 640 new augmented images were generated. The perspective intensity parameter λ was selected from 0.1 to 0.3. The number of images in different data sets is shown in Table 1.

3.2. Implementation Details during Training Process. All experiments were performed on a personal computer: Lenovo R720 (a Core i7-7700HQ CPU @ 2.80 GHz, 8 GB DDR4 memory, and 2 GB memory NVIDIA GeForce GTX 1050 Ti GPU). All the training and testing processes were conducted on the GPU. The YOLOv3 neural network was developed using Python 3.6.5 under TensorFlow 1.8.0 frame.

Before the beginning of experiments, the k -means clustering algorithm was applied on the size of the bounding boxes of images in DA_{OR} to obtain the bounding box priors and facilitate network learning and detection results. The clustering results are shown in Figure 6. The number of clusters is set at 9 as follows: (15 × 11), (13 × 14), (20 × 17), (22 × 21), (25 × 26), (35 × 31), (45 × 45), (74 × 74), and (131 × 160).

In order to improve the detection accuracy of the detector and adapt to the required input format of the Darknet53, the size of the input image is set to 416 × 416 pixels. Due to

FIGURE 7: Sample images of detected bolt delayed fracture.

the constraints of the GPU memory, the batch size was set to 8. And the training step was set to 10000 to analyze the training process by the loss curve. To prevent training the neural network from scratch, the internal adjustable parameters were initialized by using a pretrained weight, which can be obtained from the website (<https://pjreddie.com/yolo/>). The initial learning rate was set to 0.003 through trial and error with the help of the validation set. The λ_{coord} and λ_{noobj} are set to 5 and 0.5, respectively. To analyze the impact of different data augmentation methods, six neural networks were trained using six training sets. The parameter settings during training process are the same except using the different training sets.

4. Result and Discussion

After training, the images in the testing set were used to test the performance of the six detectors, and the IOU threshold is set to 0.5, 0.6, and 0.7. The AP values of six detectors under different IOU thresholds were calculated, as shown in Table 2. The highest AP value is 89.14%, which indicates that the trained detector has a strong generalization and excellent detection performance, and the AP value decreases

with the increase of the IOU threshold. The AP of the detector trained using DA_{OR} is used as a benchmark, and the AP increment of other detectors is used to estimate the usefulness of different methods. The PT and FL both improve the AP value on the testing set, and the highest increment of AP is induced by PT, achieving 4.52%, 13.39%, and 18.45% corresponding to three different IOU thresholds. In theory, five data augmentation methods all can improve the richness of the training set, and the performance of the detectors trained by augmented training sets should be better than the detector trained by the original training set. However, the BT, GB, and SC reduce the performance, as shown in Table 2. The reason can be that although the change of lighting intensity, distance, and resolution was considered during image collection, the number of the collected raw images in the testing set is too small to represent the entire image sample. Hence, the promotion of the ability of the detector to detect vague images, brightness changes, and resolution changes cannot be reflected on the existing testing set, whereas the improvement of the ability to detect objects captured from different viewpoints is the most obvious, because all images were captured from different viewpoints.

The detection results of some images in the testing set are shown in Figure 7. The fractured bolts in the image were

FIGURE 8: Sample images of detected bolt delayed fracture considering the color and weather.

FIGURE 9: Detected bolt delayed fracture on an actual bridge.

automatically detected (indicated with a solid red box) by the detector after the test images were inputted into the best detector. The detection process spends only 0.06 seconds for each input image (416×416 resolution). It should be noted that the detection speed is affected by hardware limitations. In [41], the detection speed for a 416×416 resolution image is 0.029 seconds. Hence, this method can accomplish real-time autonomous damage detection when a camera is used in conjunction with a processor. The proposed method can facilitate the transition from manual inspection to automated inspection or monitoring carried out by fixed cameras, UAVs, or remote-controlled robots in the future.

The generalization ability of the trained detector was further demonstrated using the new images of bolts with different colors (black, red, gray, and blue) and covered with raindrops and the images of fractured bolts from an actual bridge (two 3024×4032 -pixel images taken from a real long-span steel bridge in China). The detection results are shown in Figures 8 and 9, and the trained detector can correctly detect the damage from the new images. The results show that the trained detector does not overfit the two sample steel plates. It also demonstrates the practicality of the proposed method.

On the other hand, although the detector can detect the damage correctly, the predicted bounding boxes do not perfectly fit the fractured bolts. The minor errors may be induced by the limitation of the training set, such as the lack of images taken from actual bridges. In addition, a comprehensive analysis of the effectiveness of different augmentation techniques for this detection task needs a comprehensive image dataset. The images in the dataset should be collected from actual engineering. Thus, more actual images need to be collected and a larger image dataset will be established in the future to further analyze the effectiveness of different augmentation methods and how to use them in combination.

5. Conclusion

This paper presents a new, automated method to inspect fracture failures for bolts. The method is developed based upon the CNN-based object detection algorithm YOLOv3,

and the performance of the detector is improved by data augmentation. An image dataset was developed through image acquisition, image labeling, and data augmentation, and six YOLOv3 neural networks were trained using different augmented training sets to analyze the impact of different augmentation methods. The highest AP of the trained detectors is 89.14% when the IOU threshold equals to 0.5. The effectiveness of different data augmentation methods is evaluated by the increment of AP. The highest increment of AP on the testing set is achieved by perspective transformation augmentation. The detection speed of the trained detector achieved 0.06 seconds for each input image with 416×416 resolution. The generalization of the trained network and the practicality of the proposed method were validated using new images that were never used in the training and testing. The proposed method has the potential to enable safe, real-time, and autonomous detection of delayed fracture of high-strength bolts with high accuracy.

Data Availability

The datasets, codes, and weight files used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the General Project of the Natural Science Foundation of China (Grant 51778111), the Fundamental Research Funds for the Central Universities (Grant DUT19TD26), the General Project of Natural Science Foundation of Jiangsu Province of China (Grant BK20181198), and the Dalian High Level Talent Innovation Support Program (Grant 2019RD01).

References

- [1] D. Pines and A. E. Aktan, "Status of structural health monitoring of long-span bridges in the United States," *Progress in Structural Engineering & Materials*, vol. 4, no. 4, pp. 372–380, 2002.
- [2] J. Peng, L. Xiao, J. Zhang, C. S. Cai, and L. Wang, "Flexural behavior of corroded HPS beams," *Engineering Structures*, vol. 195, pp. 274–287, 2019.
- [3] L. Xiao, J. Peng, J. Zhang, Y. Ma, and C. S. Cai, "Comparative assessment of mechanical properties of HPS between electro-chemical corrosion and spray corrosion," *Construction and Building Materials*, vol. 237, article 117735, 2020.
- [4] W. Wang, X. Wang, X. Hua, G. Song, and Z. Chen, "Vibration control of vortex-induced vibrations of a bridge deck by a single-side pounding tuned mass damper," *Engineering Structures*, vol. 173, pp. 61–75, 2018.
- [5] X. Yin, G. Song, and Y. Liu, "Vibration suppression of wind-traffic/bridge coupled system using multiple pounding tuned mass dampers (MPTMD)," *Sensors*, vol. 19, no. 5, p. 1133, 2019.
- [6] J. Zhou, L. Huo, G. Song, and H. Li, "Deep learning-based visual inspection for the delayed brittle fracture of high-strength bolts in long-span steel bridges," in *International Conference on Image and Video Processing, and Artificial Intelligence*, p. 1132129, Shanghai, China, 2019.
- [7] X. Ye, C. Dong, and T. Liu, "A review of machine vision-based structural health monitoring: methodologies and applications," *Journal of Sensors*, vol. 2016, no. 5, Article ID 7103039, 2016.
- [8] Y. Tan and L. M. Zhang, "Computational methodologies for optimal sensor placement in structural health monitoring: a review," *Structural Health Monitoring-an International Journal*, vol. 19, no. 4, pp. 1287–1308, 2020.
- [9] M. Abdulkarem, K. Samsudin, F. Z. Rokhani, and M. F. A. Rasid, "Wireless sensor network for structural health monitoring: a contemporary review of technologies, challenges, and future direction," *Structural Health Monitoring-an International Journal*, vol. 19, no. 3, pp. 693–735, 2020.
- [10] L. Huo, H. Cheng, Q. Kong, and X. Chen, "Bond-slip monitoring of concrete structures using smart sensors—a review," *Sensors*, vol. 19, no. 5, p. 1231, 2019.
- [11] T. Wang, G. B. Song, S. P. Liu, Y. R. Li, and H. Xiao, "Review of bolted connection monitoring," *International Journal of Distributed Sensor Networks*, vol. 9, no. 12, 2013.
- [12] Z. Zhao, P. Chen, E. Zhang, and G. Lu, "Health monitoring of bolt looseness in timber structures using PZT-enabled time-reversal method," *Journal of Sensors*, vol. 2019, 8 pages, 2019.
- [13] T. Jiang, Q. Wu, L. Wang, L. Huo, and G. Song, "Monitoring of bolt looseness-induced damage in steel truss arch structure using piezoceramic transducers," *IEEE Sensors Journal*, vol. 18, no. 16, pp. 6677–6685, 2018.
- [14] F. Wang, L. S. Huo, and G. Song, "A piezoelectric active sensing method for quantitative monitoring of bolt loosening using energy dissipation caused by tangential damping based on the fractal contact theory," *Smart Material Structures*, vol. 27, no. 1, article 015023, 2017.
- [15] D. Chen, L. Huo, and G. Song, "EMI based multi-bolt looseness detection using series/parallel multi-sensing technique," *Smart Structures and Systems*, vol. 25, no. 4, pp. 423–432, 2020.
- [16] Y. Liang, Q. Feng, D. Li, and S. Cai, "Loosening monitoring of a threaded pipe connection using the electro-mechanical impedance technique—experimental and numerical studies," *Sensors*, vol. 18, no. 11, p. 3699, 2018.
- [17] E. E. Ungar, "The status of engineering knowledge concerning the damping of built-up structures," *Journal of Sound and Vibration*, vol. 26, no. 1, pp. 141–154, 1973.
- [18] A. Milanese, P. Marzocca, J. M. Nichols, M. Seaver, and S. T. Trickey, "Modeling and detection of joint loosening using output-only broad-band vibration data," *Structural Health Monitoring-an International Journal*, vol. 7, no. 4, pp. 309–328, 2008.
- [19] L. Huo, D. Chen, Q. Kong, H. Li, and G. Song, "Smart washer—a piezoceramic-based transducer to monitor looseness of bolted connection," *Smart Materials and Structures*, vol. 26, no. 2, article 025033, 2017.
- [20] L. S. Huo, D. D. Chen, Y. B. Liang, H. N. Li, X. Feng, and G. B. Song, "Impedance based bolt pre-load monitoring using piezoceramic smart washer," *Smart Materials and Structures*, vol. 26, no. 5, p. 057004, 2017.
- [21] N. Zhao, L. Huo, and G. Song, "A nonlinear ultrasonic method for real-time bolt looseness monitoring using PZT transducer-enabled vibro-acoustic modulation," *Journal of Intelligent Material Systems and Structures*, vol. 31, no. 3, pp. 364–376, 2020.
- [22] Y. Hagihara, "Evaluation of delayed fracture characteristics of high-strength bolt steels by CSRT," *ISIJ International*, vol. 52, no. 2, pp. 292–297, 2012.
- [23] E. Akiyama, "Evaluation of delayed fracture property of high strength bolt steels," *ISIJ International*, vol. 52, no. 2, pp. 307–315, 2012.
- [24] B. F. Spencer, V. Hoskere, and Y. Narazaki, "Advances in computer vision-based civil infrastructure inspection and monitoring," *Engineering*, vol. 5, no. 2, pp. 199–222, 2019.
- [25] C. Z. Dong and F. N. Catbas, "A review of computer vision-based structural health monitoring at local and global levels," *Structural Health Monitoring-an International Journal*, vol. 20, no. 2, pp. 692–743, 2021.
- [26] J. H. Park, T. C. Huynh, S. H. Choi, and J. T. Kim, "Vision-based technique for bolt-loosening detection in wind turbine tower," *Wind and Structures*, vol. 21, no. 6, pp. 709–726, 2015.
- [27] Y. J. Cha, K. You, and W. Choi, "Vision-based detection of loosened bolts using the Hough transform and support vector machines," *Automation in Construction*, vol. 71, pp. 181–188, 2016.
- [28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [29] B. Du, Y. He, Y. He, J. Duan, and Y. Zhang, "Intelligent classification of silicon photovoltaic cell defects based on eddy current thermography and convolution neural network," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6242–6251, 2020.
- [30] Y. Xu, M. Fu, Q. Wang et al., "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 4, pp. 1452–1459, 2021.
- [31] X. H. Yuan, J. F. Shi, and L. C. Gu, "A review of deep learning methods for semantic segmentation of remote sensing imagery," *Expert Systems with Applications*, vol. 169, article 114417, 2021.

- [32] T.-C. Huynh, J.-H. Park, H.-J. Jung, and J.-T. Kim, "Quasi-autonomous bolt-loosening detection method using vision-based deep learning and image processing," *Automation in Construction*, vol. 105, article 102844, 2019.
- [33] X. F. Zhao, Y. Zhang, and N. N. Wang, "Bolt loosening angle detection technology using deep learning," *Structural Control & Health Monitoring*, vol. 26, no. 1, article e2292, 2019.
- [34] C. Y. Wang, N. Wang, S. C. Ho, X. M. Chen, and G. B. Song, "Design of a new vision-based method for the bolts looseness detection in flange connections," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 2, pp. 1366–1375, 2020.
- [35] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, USA, 2014.
- [36] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, Santiago, USA, 2015.
- [37] S. Q. Ren, K. M. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [38] W. Liu, D. Anguelov, D. Erhan et al., "SSD: single shot multi-box detector," in *European Conference on Computer Vision*, pp. 21–37, Amsterdam, Netherlands, 2016.
- [39] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Seattle, USA, 2016.
- [40] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, Honolulu, USA, 2017.
- [41] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," 2018, <http://arxiv.org/abs/1804.02767>.
- [42] A. Bochkovskiy, W. Chien-Yao, and H. Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection arXiv," 2020, <http://arxiv.org/abs/2004.10934>.
- [43] Y. N. Tian, G. D. Yang, Z. Wang, H. Wang, E. Li, and Z. Z. Liang, "Apple detection during different growth stages in orchards using the improved YOLO-V3 model," *Computers and Electronics in Agriculture*, vol. 157, pp. 417–426, 2019.
- [44] S. S. Kumar, M. Z. Wang, D. M. Abraham, M. R. Jahanshahi, T. Iseley, and J. C. P. Cheng, "Deep learning-based automated detection of sewer defects in CCTV videos," *Journal of Computing in Civil Engineering*, vol. 34, no. 1, article 04019047, 2020.
- [45] D. M. Han, Q. G. Liu, and W. G. Fan, "A new image classification method using CNN transfer learning and web data augmentation," *Expert Systems with Applications*, vol. 95, pp. 43–56, 2018.
- [46] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 48, 2019.
- [47] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [48] K. Wang, B. Fang, J. Y. Qian, S. Yang, X. Zhou, and J. Zhou, "Perspective transformation data augmentation for object detection," *IEEE Access*, vol. 8, pp. 4935–4943, 2020.
- [49] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.