Received 26 July 2020; | Revised 26 July 2020; | Accepted 26 July 2020

DOI: xxx/xxxx

ORIGINAL ARTICLE

Autonomous detection of damage to multiple steel surfaces from 360° panoramas using deep neural networks

Cai Luo*[1] | Leijian Yu[2] | Jiaxing Yan[3,4] | Zhongwei Li[1] | Peng Ren[1] | Xiao Bai[5] | Erfu Yang[2] | Yonghong Liu[6]

[1]College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao, China
[2]Department of Design, Manufacturing & Engineering Management, University of Strathclyde, Glasgow, U.K.
[3]School of Mechanical Engineering, Xi'an Jiaotong University, Shaanxi, China
[4]State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Shaanxi, China
[5]School of Computer Science and Engineering, Beijing Advanced Innovation Center for Big Data and Brain Computing, Beijing, China
[6]College of Mechanical and Electronic Engineering, China University of Petroleum (East China), Qingdao, China

***Correspondence**
Cai Luo, College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao, China, 266580.
Email: luo_cai@upc.edu.cn

Summary

Structural health assessments are essential for infrastructure. By using an autonomous panorama vision-based inspection system, the limitations of the human cost and safety factors of previously time-consuming tasks have been overcome. The main damage detection challenges to panorama images are (1) the lack of annotated panorama defect image data, (2) detection in high-resolution images, and (3) the inherent distortion disturbance for panorama images. In this paper, a new PAnoramic surface damage DEtection Network (PADENet) is presented to solve the challenges by (a) using an unmanned aerial vehicle (UAV) to capture panoramic images and a distorted panoramic augmentation method to expand the panoramic dataset, (b) employing the proposed multiple projection methods to process high-resolution images, and (c) modifying the faster region-based convolutional neural network (Faster R-CNN) and training via transfer learning on VGG-16, which improves the precision for detecting multiple types of damage in distortion. The results show that the proposed method is optimal for surface damage detection.

1 | INTRODUCTION

Under harsh working environments, long-term utilization or lack of maintenance, multiple regions of surface damage appear and alert us to the life span, safety and workability of civil infrastructures (Huang & Pan, 2015; Jiang & Adeli, 2010; Koch, Zhu, Paal, & Brilakis, 2015), as shown in Figure 1 . The visual changes on the surface of a civil infrastructure, such as cracks, exfoliation, efflorescence and corrosion, reflect the structural health conditions by encoding damage information (Chen & Perng, 2016; Cho, Chung, & Park, 2005; Koch et al., 2014; Kumar, 2008). Note that visual inspection is currently the principal approach for maintenance and assessment. Therefore, the reliability and efficiency of inspection systems are vital to public safety and economy. The mainstream approach of regular monitoring is primarily performed through visual inspections by trained workers. However, eye-based visual inspection is inefficient and experience-dependent and can even be dangerous when the conditions of structures are hazardous (Adeli & Jiang, 2008; Agnisarman, Lopes, Madathil, Piratla, & Gramopadhye, 2019).

Hence, automatic visual inspection is highly desirable for addressing the limitations of the human-based visual method. Various approaches for efficiently assessing infrastructure have been studied (Abdel-Qader, Abudayyeh, & Kelly, 2003; Vaghefi et al., 2011). Most studies on damage detection focus on image-processing techniques (IPTs)(Fan et al., 2019; Jiang & Adeli, 2010; Uijlings, Van De Sande, Gevers, & Smeulders, 2013). With this type of computer vision-based method, some specific types of damage, such as cracks, concrete spalling

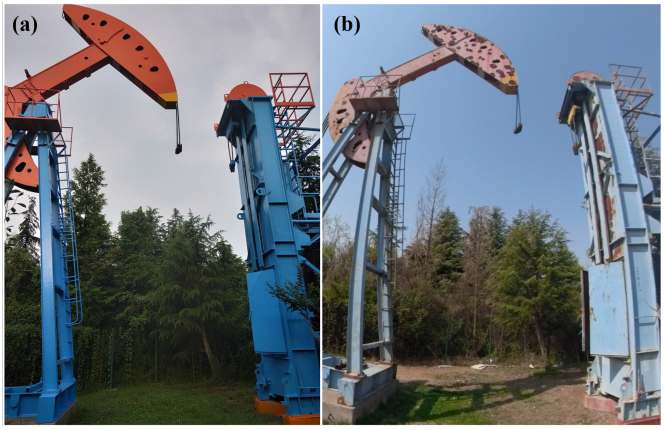

FIGURE 1 High-rise structure exhibiting apparent defects on the surface after several years of outdoor service. (a) Infrastructure in good condition. (b) Infrastructure in poor condition after several years of service.

and steel defects, can be detected (Abdel-Qader et al., 2003; Chen & Perng, 2016). Furthermore, automatic damage detection algorithms have been successfully utilized in autonomous moving platforms (Y.-F. Liu, Nie, Fan, & Liu, 2019; Morgenthal & Hallermann, 2014; Zhou, Li, & Cheng, 2005).

Although previous researchers have demonstrated various methods for autonomous inspection (Ahmadlou & Adeli, 2010; Alam, Siddique, & Adeli, 2020; Pereira, Piteri, Souza, Papa, & Adeli, 2020; Rafiei & Adeli, 2017a), the current inspection approaches cannot be utilized to assess images taken by panoramic cameras. Damage detection in 360° images is challenging in three aspects. First, there are no public panoramic datasets available. Previous researchers used their own datasets to evaluate panoramic approaches (Deng, Zhu, & Ren, 2017; Fan et al., 2019; Lenjani, Yeum, Dyke, & Bilionis, 2020; Y. Li, Tong, et al., 2019). Second, the panoramic pictures containing damage account for only a small fraction of the images that are analyzed, which have cluttered backgrounds (Cha, Choi, Suh, Mahmoudkhani, & Büyüköztürk, 2018; Jang, An, Kim, & Cho, 2020; Zhang, Xiao, & Yang, 2017). Third, the dataset quality significantly depends on the flight conditions and time of day. The season and weather also influence the quality of the images taken by drones (Kang & Cha, 2018). Therefore, visual inspection of the condition of construction systems plays a crucial role in avoiding vital accidents and ensuring safe construction operations.

In response to these challenges, we propose a method that employs a deep neural network for detecting various damage to multiscale steel surfaces in panoramic images.

The key technical contributions of this article are summarized as follows:

(1) A highly efficient PAnoramic surface damage DEtection Network (PADENet) is proposed for identifying multiple types of high-rise infrastructure surface damage from 360° images. This design helps to localize defects at various distortions and ensures a relatively high accuracy rate while improving autonomous infrastructure assessment efficiency.

(2) A distorted panoramic augmentation method is demonstrated. We show how to efficiently extend the panoramic image dataset and train the proposed method by transfer learning and knowledge sharing. With an insufficient number of training images, the novel network can retain a high accuracy with affordable computation costs.

(3) By using an unmanned aerial vehicle (UAV) equipped with a panoramic camera, we present a real industrial application scenario for our proposed method of detecting multiple types of steel surface damage from theoretical and experimental perspectives. With our autonomous steel defect detection and classification approach, workers can prepare repair equipment effectively according to the damage type.

To test the performance of the proposed neural network, comprehensive validation tests are conducted. Using our collected panoramic image dataset, we evaluate the proposed method with state-of-the-art neural network object detection algorithms. The results show that our approach has high steel surface damage detection accuracy and reliably matches the ground truth of damage classification.

2 | RELATED WORK

2.1 | Infrastructure surface damage inspection using autonomous vehicles

To increase the frequency of structural health inspection while decreasing the working time and improving the working environment of human inspectors, autonomous moving platforms have been widely applied to visual inspection tasks. Previously, the visual sensors explicitly designed for damage detection were heavy and large. Only unmanned ground vehicles (UGVs) are capable of carrying these devices to assessment locations. Many successful applications of UGVs have been performed, such as (Dueholm, Kristoffersen, Satzoda, Moeslund, & Trivedi, 2016).

However, due to the restrictions of movement, there are still many places that UGVs cannot reach. For example, it is challenging to assess surface health conditions of tall civil structures, bridges, and storage areas with narrow paths without the aid of specially designed inspection equipment. Additionally,

remarkable progress has been made in flying platforms such as UAVs in recent years. Because UAVs can be deployed in isolated or hazardous sites, these vehicles are increasingly used for automotive visual inspection works.

2.2 | Panoramic image dataset of infrastructure surface damage

Recent automotive visual inspection works, such as damage detection, corrosion segmentation, stereo-road inspection, and monocular depth estimation, have mainly focused on the study of downward- and forward-facing platform-mounted vision sensors. To be able to cover the entire assessment area, an inspector needs to plan a go-return routine for UAVs due to the limitation of the view angle of the camera. This requirement severely increases the inspection time and decreases the efficiency of an autonomous assessment system.

To overcome the potential effects of the limited view angle, future autonomous inspection vehicles will require a 360° inspection ability for reliability and efficiency across clustered working environments. Equipped with panoramic visual sensors, UAVs are able to detect damage in 360° without rotating the cameras. With the recent advances in panoramic sensors, UGVs equipped with digital cameras provide new choices for infrastructure visual assessment, for example, as described in (Bertozzi, Castangia, Cattani, Prioletti, & Versari, 2015; Y. Li, Tong, et al., 2019). Although the aforementioned UGV-based detection system can recognize multiple objects, annotated 360-degree datasets for damage detection are lacking. Additionally, UGVs cannot reach many locations, which are more likely to be inaccessible for regular visual inspection. Therefore, the use of UAVs for 360-degree visual inspection is becoming increasingly important (Fan et al., 2019; Kang & Cha, 2018).

2.3 | Deep neural network damage detection

Previous studies mainly used pre- and postprocessing methods to detect specific types of structural damage. For example, Chun proposed a pavement crack detection algorithm that uses a naive Bayes machine learning method (Chun, Hashimoto, Kataoka, Kuramoto, & Ohga, 2015). Abdel-Qader showed that fast Harr transfer (FHT) performed well for the detection of cracks in bridges (Abdel-Qader et al., 2003). A nonlinear dynamics measure based on the fractal dimension was validated for quantifying damage in high-rise buildings (Amezquita-Sanchez & Adeli, 2015). Loosened bolts can be detected by the methods proposed by German, Brilakis, & DesRoches (2012). Potholes in asphalt pavement were analyzed in the work of Koch & Brilakis (2011). However, IPT-based visual inspection methods can only detect one specific type of damage, and noise in real applications, such as illumination and distortion, greatly affects the detection results. To optimize the robustness and diversity of IPT-based algorithms in industry applications, machine learning approaches, such as AdaBoost, the K-nearest neighbors algorithm and the restricted Boltzmann machine, have been used to find the desired solution for defect detection. For example, the deep Boltzman machine was proposed to estimate concrete compression strength based on mixture proportions, and the model demonstrates more accurate results than the back-propagation neural network and support vector machine methods (Rafiei, Khushefati, Demirboga, & Adeli, 2017). To extract hidden features in high-rise building structures, a neural dynamic classification algorithm was proposed to detect the global health of the structure (Rafiei & Adeli, 2017b). The accuracy of this method was tested in a 3D 1:20 scaled building structure, and the results showed that this method improved the detection accuracy. However, these algorithms are suboptimal for handling images with cluttered backgrounds.

The advent of deep learning approaches is promising for addressing these limitations (Cha et al., 2018; Gao & Mosalam, 2018; Liang, 2019; Narazaki et al., 2020; Ni, Zhang, & Chen, 2019; Rafiei & Adeli, 2018; Sajedi & Liang, 2020). Convolutional neural networks (CNNs) have attracted much attention because they can detect multiple damage types (Ren, He, Girshick, & Sun, 2015; Uijlings et al., 2013). CNNs have been used to assess concrete cracks, road cracks and other types of damage. Undeniably, these works achieve excellent performance in real-world applications.

3 | METHODOLOGY

This section presents a novel three-step methodology for detecting surface damage in panoramic images of tall structures. The first step includes the following tasks: (a) merging of the front and back fisheye images into panoramic images and (b) stereo-projection of the panoramic image into four subwindows. The second step includes the following tasks: (a) transfer learning, (b) region proposal network (RPN) training, and (c) Fast R-CNN training. The third step includes the following tasks: (a) bounding box alignment and (b) soft selection.

3.1 | Spherical panoramic to planar projection

Compared with traditional forward-facing planar images, which record only a small portion of the 3D scene, panoramic images capture all the virtual information of the surrounding world. Panoramic images eliminate the need to take a series of pictures at a 360° rotation.

Panoramic vision sensors provide an ultrawide view of the angle by bending the incident light. To use both the front and back images at the same time, the two distorted spherical images must be converted into an ordinary planar image, as shown in Figure 2 .

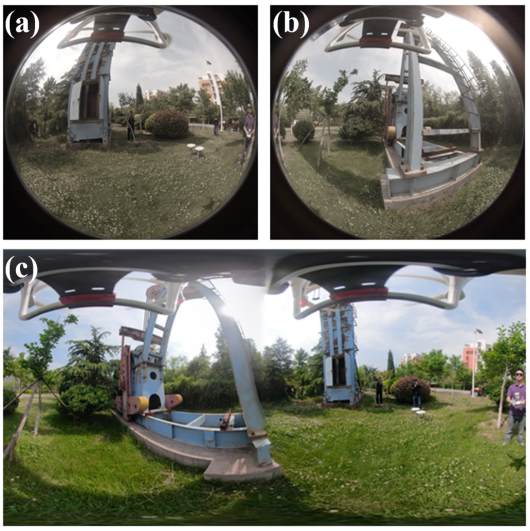

FIGURE 2 Merging the front and back images into a 360° panoramic image. (a) and (b) Original front and back fisheye images, respectively. (c) Generated panoramic image.

In the tests, we use a circular fisheye projection model to accomplish the equirectangular transfer tasks as illustrated in Figure 3 . A fisheye transformation is necessary to obtain a natural outcome. The algebraic form (Yang, Qian, Kämäräinen, Cricri, & Fan, 2018) of the fisheye projection can be described as follows:

$$\begin{aligned} x_p &= \frac{d+1}{d+\mathrm{R}\cos(\phi)}\mathrm{R}\sin(\phi) \\ y_p &= \frac{d+1}{d+\mathrm{R}\cos(\theta)}\mathrm{R}\sin(\theta) \end{aligned} \tag{1}$$

where R denotes the spherical radius and d is the equirectangular projection center.

3.2 | Multiple surface damage classifier

Faster R-CNN is the most popular detector for detecting and localizing multiple types of surface damage (Girshick, 2015; Ren et al., 2015). The architecture of this work is inspired by Faster R-CNN, and its neural network is modified to automatically identify surface defects in panoramas in our proposed algorithm, which is explained in detail in the following section.

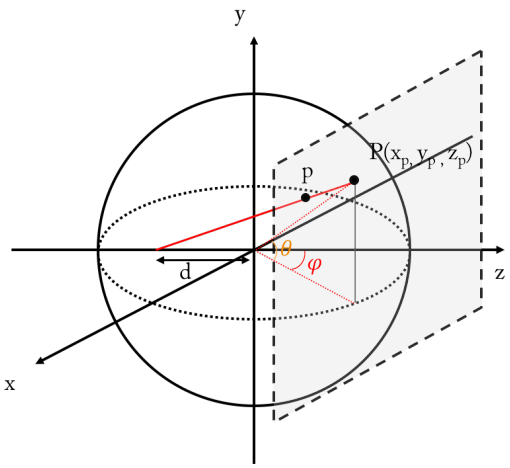

FIGURE 3 Fisheye projection (Yang et al., 2018).

3.2.1 | Layers in the deep learning model

The aforementioned Faster R-CNN is composed of an RPN and a Fast R-CNN that share the same CNN to capture object features. In this section, the main layer of the CNN, the convolutional layer, is presented. A convolutional layer performs the convolution operation through an input array. Compared with classical neural networks with full connectivity, the units called the filter or kernel are sparsely connected. The convolutional layer performs unit-by-unit multiplications between the input array and the filter, where the filter slides across the input array with a predefined step size of s. The number of multiplication operators is significantly decreased by employing the aforementioned sparse connections. The rectified linear unit (ReLU) is implemented after the convolution results. As an activation function, the ReLU facilitates faster computational speed than traditional functions such as sigmoid, tanh and arctan.

To further reduce the spatial size of an input array, a max pooling layer is usually added after ReLU processing. Max pooling calculates the maximum value from an input array's subarrays. This process performs downsampling operations that reduce the computational burden and achieve some special functions, such as decreasing overfitting and reducing distortion.

To classify the input array, a softmax layer is added to the last layer of the neural network processing line. The softmax layer normally takes features from a FC layer that connects to all cells in the previous max pooling layer. The softmax layer calculates the probabilities of the damage type and outputs the strongest prediction.

3.2.2 | Architecture of the region proposal network

The goal of an RPN is to propose a region where an object can be potentially located. As shown in Figure 4 , the RPN takes input feature maps of any size and outputs a rectangular object proposal region. The RPN uses the aforementioned shared CNN to extract a feature map and slide another convolutional layer across the map. The RPN reshaping, followed by the ReLU, is then used to feed the features into the regression and softmax layers. Another key reason to use an RPN as the backbone of our network is the advantage of weight knowledge sharing between the RPN and the Fast R-CNN detector which can reduce computational costs significantly (Cha et al., 2018)

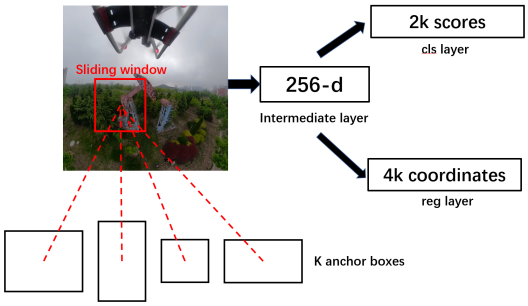

FIGURE 4 Architecture of the region proposal network.

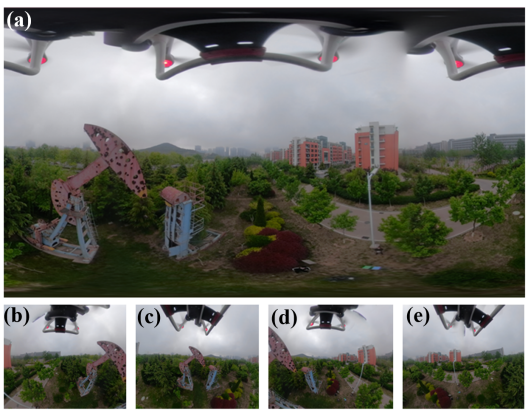

FIGURE 5 Multiprojection processing. (a) Generated 360° panoramic image. (b-e) To compensate for the distortion in stereographic projections, four stereographic images with 180° horizontal and vertical spans are adopted.

TABLE 1 The detail specification of the proposed PADENet layers.

Layer	Type	Depth	Size	Stride	Paddings
1	Conv1_1	64	3	1	SAME
2	Conv1_2	64	3	1	SAME
3	Max pooling1	64	2	2	VALID
4	Conv2_1	128	3	1	SAME
5	Conv2_2	128	3	1	SAME
6	Max pooling2	128	2	2	VALID
7	Conv3_1	256	3	1	SAME
8	Conv3_2	256	3	1	SAME
9	Conv3_3	256	3	1	SAME
10	Max pooling3	256	2	2	VALID
11	Conv4_1	512	3	1	SAME
12	Conv4_2	512	3	1	SAME
13	Conv4_3	512	3	1	SAME
14	Max pooling4	512	2	2	SAME
15	Conv5_1	512	3	1	SAME
16	Conv5_2	512	3	1	SAME
17	Conv5_3	512	3	1	SAME
18	Max pooling5	512	2	2	SAME
19	Conv6_1	512	3	1	SAME
20	Conv6_2	512	3	1	SAME
21	Conv6_3	512	3	1	SAME
22	RoI Pooling	-	-	-	-
23	Fully connected_1	4096	-	-	-
24	Dropout_1	-	-	-	-
25	Fully connected_2	4096	-	-	-
26	Dropout_2	-	-	-	-
27	Fully connected_3	5	-	-	-
26	Softmax & Regressor	-	-	-	-

TABLE 2 Specifications of the proposed RPN layers.

Layer	Type	Depth	Size	Stride	Paddings
1-13	Shared CNN layers as shown in Table 1	-	-	-	-
14	Add_1	512	-	-	-
15	ConvRPN1_1	256	1	1	SAME
16	ConvRPN1_2	512	3	1	SAME
17	ConvRPN1_3	256	1	1	SAME
18	ConvRPN1_4	512	3	1	SAME
19	ConvRPN1_5	54	1	1	VALID
20	RPN reshape_1	-	-	-	-
21	Softmax & Regressor	-	-	-	-

3.3 | Architecture of the panoramic surface damage detection network

Our goal is to propose a machine learning algorithm that can process full panoramic images. By using panoramic images, our method can detect surface defects in 360° from all angles. Traditional methods are unable to detect distorted surface defects. Inspired by the works of Wenyan Yang et al. (Yang, Qian, Kämäräinen, Cricri, & Fan, 2018), we propose a multi-projection approach with a Faster R-CNN-based PADENet to

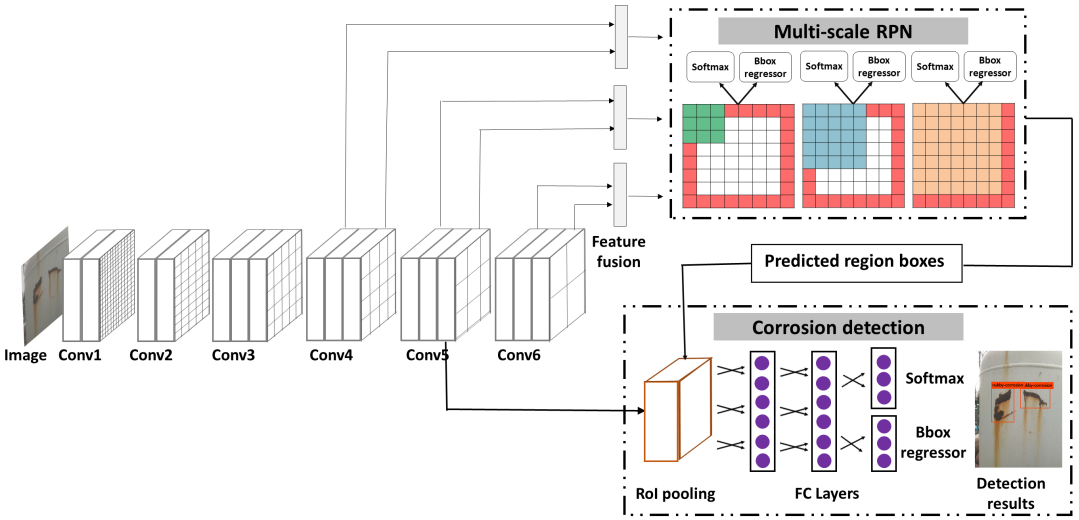

FIGURE 6 Schematic architecture of PADENet.

overcome the problem of distortion and conduct a structural health assessment (Yang et al., 2018). In the multiprojection step, to compensate for the distortion in stereographic projections, four stereographic images with 180° horizontal and vertical spans, which means that the overlap between each neighboring image in the horizontal direction is 90°, are used, as shown in Figure 5 .

After the multiprojection step, each subwindow uses the shared bottom layers to compute the feature maps and employs PADENet to detect defects. The architecture of PADENet is shown in Figure 6 . Three scale-sensitive Fast R-CNN networks are proposed to detect defects of different scales and aspect ratios. In the case of long and thin defects, the region of interest (RoI) pooling region likely has few pixels, which makes these defects difficult to predict and classify. To overcome this challenge, we adopt a straightforward strategy with a multiscale region proposal network (M-RPN) to improve the precision for long and thin defects. Based on their percentages of coverage on the feature map, the groups are named small, medium, and large. To generate feature maps for different scales, the Conv6 module is added to VGG 16. Moreover, feature fusion is adopted to improve the representation of the feature maps. This process improves the detection precision for multiscale images of defects with a complicated background (R. Li, Yuan, Zhang, & Yuan, 2018; Ren et al., 2015).

Table 1 shows the details of the proposed PADENet's layers. Additionally, one of the RPNs used in this study is shown in Table 2 . The main difference among the three RPNs is the number of CNN layers shared with Fast R-CNN. RPN_2 shares 1-17 layers as shown in Table 1 , while RPN_3 shares 1-21 layers. The RPN aims to generate potential bounding boxes for object detection, and it can influence the performance of the detector significantly. Thus, in addition to Conv5, Conv6 is added to generate more potential bounding boxes based on the feature map, which contains more abstract or global features. Moreover, Conv4 is also used to generate potential bounding boxes based on the feature map, which contains more simple features. All of these bounding boxes proposed by the 3 RPNs will be processed by the RMS to generate more reliable bounding boxes compared to a single RPN. Additionally, compared to Conv5, Conv6 has high convolutional layers and contains more abstract (global) features. The high convolutional layers will compress the features of small steel defects so that these small defects are ignored by the detection network. However, we still want to detect small steel defects. Moreover, receptive fields will also influence the performance of the detector(Y. Li, Chen, Wang, & Zhang, 2019). ZFNet (Zeiler & Fergus, 2014), VGGNet (Simonyan & Zisserman, 2014) and ResNet (He, Zhang, Ren, & Sun, 2016) are popular backbones for the detection of objects that have the same receptive field of the last convolutional layer. Thus, we follow their structure and use the outputs from Conv5 rather than Conv6 to maintain the same receptive fields. Conv6 aims to generate a bounding boxing at a different scale to improve the performance of the detector. Padding is applied to the input image so that the input image is fully covered by the filter, as indicated by "SAME". When the stride is 1, the output data are the same size as the input data. With the "VALID" indicator, there is no padding applied to the input image. In this case, the filter stays at the valid position inside the input image.

To obtain a well-trained CNN, a sufficient amount of training data is a prerequisite. However, a large annotated damage dataset is difficult to obtain for this type of research due to the human resource requirements needed for annotation (Bang,

Park, Kim, & Kim, 2019). To accelerate the convergence of the network and avoid overfitting the proposed method with a limited amount of data, the transfer learning method is used to pretrain the proposed CNN by using sufficient data from other domains. Since there is no need to start from randomly initialized neural network weights, CNNs with transfer learning have been successfully used in the fields of construction tools, pavement defect detection and plant disease detection (R. Li et al., 2018; S. Li, Zhao, & Zhou, 2019).

TABLE 3 Numbers and Classification standard for steel surface damage.

Classes	Numbers	Definition
Fastener	4,380	Bolt and nut corrosion
Exfoliation	3,160	Cracked coating
Nubby	5,270	Aspect ratio of less than 1:2
Bar	3,190	Aspect ratio of more than 1:2

4 | PROPOSED DATASET AND IMPLEMENTATION DETAILS

4.1 | Building dataset

To develop a panoramic image dataset containing bar corrosion, fastener corrosion, nubby corrosion and exfoliation, 2270 images (with a resolution of 4992 × 2496 pixels) are collected using a GoPro Fusion camera.

We mount the camera under the UAV as shown in Figure 7 and shoot six image bursts every two seconds. Images are taken during spring and summer under various illumination conditions at midday and sunset times at three different locations in an aging industrial factory that is equipped with containers, an oil derrick, drilling equipment, a cabinet, transmission lines and steel scaffolding located in Tsingtao, China. The first location is an outside storage place near the seaside, where the surface of gas/oil containers is much easier to be damaged by wind erosion. The second location is an industrial operation area with high construction devices. It provides a complex building combination for our detection algorithm, which includes an oil derrick, drilling equipment and a power distribution cabinet. The third location is on the far side of the factory, with an abundance of trees, bushes, transmission lines, and an unattended steel scaffolding. The trees and bushes block parts of the surface defects, and some of the leaves of the trees have a similar color to the corrosion during autumn, which are challenges to surface damage detection algorithms. The attended devices can be employed as a platform that we can use to apply artificial corrosion on surfaces in order to assess the robustness of the steel surface defect detection algorithm. Considering that the classification performance may be affected by the camera's view angle, several images are taken from multiple view angles. A damage region aspect ratio of less than 1:2 in the images is considered to be nubby corrosion. Corrosion is treated as bar corrosion when the aspect ratio of the defect region is more than 1:2. The fastener corrosion mainly includes bolt and nut corrosion, and cracked coatings are treated as exfoliation corrosion.

The ground truths of the captured images are annotated manually using LabelImg software. These labels are saved as

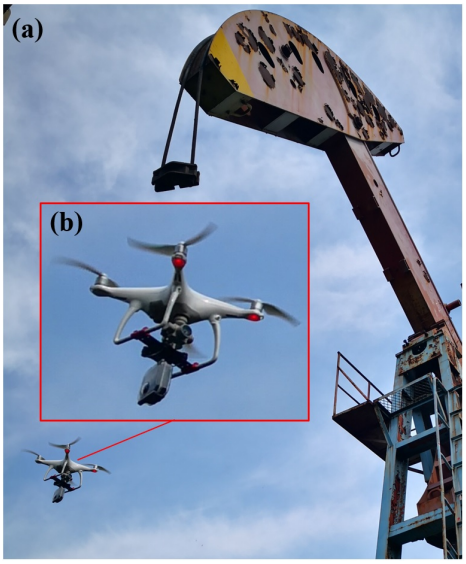

FIGURE 7 Panoramic camera carried by a UAV to collect surface data from tall structures. (a) A UAV with a panoramic sensor is collecting high-altitude image data. (b) Enlarged UAV image that shows that a panoramic sensor is attached to the landing structure of the UAV.

XML files in PASCAL VOC format (Everingham, Van Gool, Williams, Winn, & Zisserman, 2010). In the labeling process, a surface damage area smaller than 4 mm^2 is ignored because very small amounts of damage are not harmful to structures, and resources are not required to repair them at this stage.

For a deep neural network, the limited supply of 360° images makes training a defect detector challenging. To effectively use the collected images and decrease the probability of overfitting, we propose a distorted data augmentation method, as shown in Figure 8 to increase the number of distorted training data. The database is further improved by an increase in the amount of training data. We first fuse the fisheye images into panoramic images. The objects at the bottom of the panoramic images are more significantly distorted when passing through the projection. Based on this characteristic, we generate distorted objects through the projection procedure. We implement an annotation tool in which the user is capable of changing

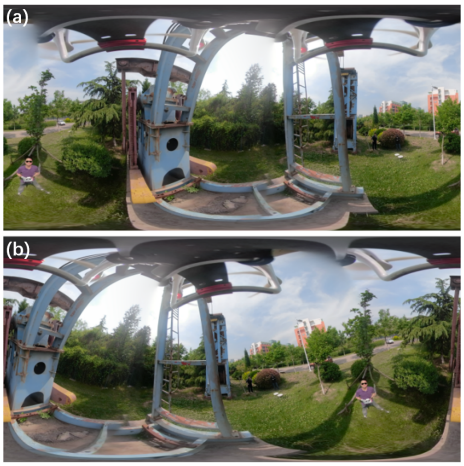

FIGURE 8 Data augmentation procedure. (a) Original image. (b) Horizontal movement for data augmentation.

the center of the field of view. During the annotation process, the annotators are asked to rotate a random chosen defect to a position near the center of the image.

The ground truths of the damage are annotated using LabelImg software. Finally, after performing multiprojection processing as shown in Figure 5 , the labels and bounding boxes for 27,042 objects are specified in 16,000 images. A total of 16,000 images are divided into 12,800 images for training and 3,200 images for testing.

5 | MODEL TRAINING

In the training procedure, the base layer is initialized with the pretrained VGG (Simonyan & Zisserman, 2014) weights on ImageNet (Krizhevsky, Sutskever, & Hinton, 2012) and the transfer learning method described in Section 3.4 is carried out. The multitask loss function used by PADENet to learn defect region proposal classification is inspired by the work of Ren et al. (2015), and it is shown as follows:

$$L_{\text{PADENet}}\left(\{p_i\},\{loc_i\}\right)=\frac{1}{N_{\text{cls}}}\sum_i L_{\text{cls}}\left(p_i,p_i^*\right)+\lambda\frac{1}{N_{\text{reg}}}\sum_i p_i^* L_{\text{reg}}\left(loc_i,loc_i^*\right) \tag{2}$$

where i denotes the anchor number and p_i indicates the defect probability. If the anchor i is labeled positive, the ground truth p_i^* is marked as 1, and p_i^* is marked as 0 for a negative anchor. The variables loc_i and loc_i^* are the vectors of the predicted bounding box and the ground truth region, respectively. N_{cls} indicates the size of the minibatch, and N_{reg} denotes the number of anchors. L_{cls} and L_{reg} are the classification and regression losses, respectively.

The L_{cls} is calculated using the log loss function as follows:

$$L_{cls}=-\frac{1}{N}\sum_{i=1}^{N} X_1\cdot\log\left(p\left(X_1\right)\right)+\left(1-X_1\right)\cdot\log\left(1-p\left(X_1\right)\right) \tag{3}$$

and the L_{reg} can be denoted as follows:

$$L_{reg}\left(X_1,X_2\right)=\begin{cases}0.5\left(X_1-X_2\right)^2 & \text{if } \left|X_1-X_2\right|<1\\ \left|X_1-X_2\right|-0.5 & \text{otherwise}\end{cases} \tag{4}$$

where X_1 and X_2 represent the sample variables.

6 | EXPERIMENTS

6.1 | Testing, validation and results

A training process, as shown in Figure 9 , is implemented to improve the generalization of PADENet. We take inspiration from a previous study (Ren et al., 2015) for training the initialized weight and bias parameters. The initial learning rate is set to 0.0002 with a weight decay of 0.0005 after 50,000 iterations. All the described works are trained with TensorFlow on a PC equipped with a Tesla V100 GPU and an Intel Xeon Gold 5120 CPU @ 2.20 GHz. It took 84 hours to train the PADENet

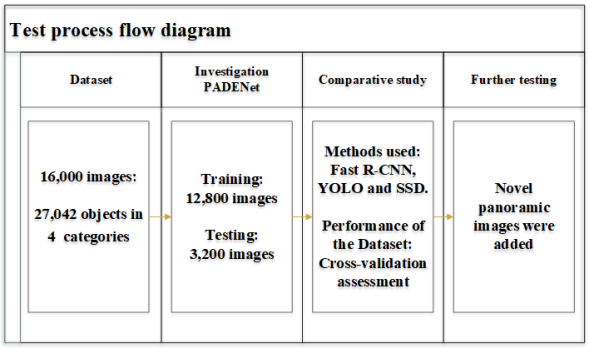

FIGURE 9 Sample images taken in different conditions with detection results and class labels. (a) Bar-corrosion detection.

with the set-up described in Section 3.3. The processing time for obtaining the detection results is 103 ms per image.

In this study, a true-positive detection is achieved if the intersection over union (IoU) is over 50%, which is the ratio of the overlap area between the suspicious zone and ground truth zone. To evaluate the surface damage detection accuracy, we apply the average precision (AP) and mean AP (mAP) as the evaluation metrics, which is the same procedure applied in the PASCAL VOC challenge. We record the AP and mAP for the four types of damage. The IoU threshold is 0.5, and the training step is 100,000.

To ensure that PADENet can detect small-scale defects, each panoramic image is separated into 4 subimages as described in Section 3.3, and then each subimage is input into the network. The outputs yield an mAP of 87.34% and APs of 92.41 88.36, 80.78, and 87.81 for fastener corrosion, exfoliation, bar corrosion and nubby corrosion, respectively. The output images are shown in Figure 10 . Images that include bar corrosion are shown in Figure 10 a1-a3. Here, different sizes of bar corrosion are successfully detected. One incident of misdetection is shown in Figure 10 a3, and it may have been related to the smaller area of corrosion relative to that in the training set. Figure 10 b1-b3 includes images of multiple types of defects under a complex background. The background includes similar colors to that of nubby corrosion and similar shapes to that of bar corrosion. As we can see from the results, the majority of defects are classified correctly and only one case of fastener corrosion is classified as nubby corrosion. Defects under various lighting conditions are demonstrated in Figure 10 c1-c3. Fastener corrosion under intense light is positively detected as well as nubby corrosion on a foggy day.

There are 57 simple damaged areas, and three different types are shown in Figure 10 . All the defect types in Figure 10 are successfully indicated and classified. The predicted damage bounding boxes match well with the ground truths.

6.2 | Comparative study with state-of-the-art detectors

The performance of the proposed PADENet approach is compared with that of different machine learning-based detectors. Figure 11 shows the precision-recall (PR) curves of the PADENet, Fast R-CNN(Girshick, 2015), you only look once (YOLO) (Redmon & Farhadi, 2018) and single-shot detector (SSD)(W. Liu et al., 2016) methods for the panoramic defect dataset.

6.2.1 | Comparison of the methods on different types of defects

Table 4 summarizes the testing results obtained using different detection methods, and Table 5 demonstrates the average GPU usage percentage in panoramic images. The effectiveness of PADENet, Faster R-CNN, YOLO and SSD was evaluated by comparing the networks trained with transfer learning. As demonstrated in Table 4 , the detection results show that the nubby corrosion and bar-corrosion performances are suboptimal due to the relatively small size of these defects, which are easily blocked with trees and buildings.

Optimal results are obtained for fastener corrosion and exfoliation defects attributed to their large size, making them difficult to block. PADENet achieves the best precision, followed by YOLO. The mAP results of Fast R-CNN are lower than those of the others, which is possibly due to its shallow backbone network and single-scale detection structure.

TABLE 4 mAP results for the evaluation of baseline surface defects in panorama steel images with the PADENet, Fast R-CNN, YOLO and SSD methods.

	PADENet	Fast R-CNN	YOLO	SSD
Fastener	92.41	82.83	85.16	86.63
Exfoliation	88.36	76.81	86.02	81.92
Nubby	87.81	81.21	82.95	81.37
Bar	80.78	69.28	75.47	72.79
mAP	87.34	77.53	82.40	80.18

TABLE 5 Average GPU usage percentage in panoramic images with the PADENet, Fast R-CNN, YOLO and SSD methods.

	PADENet	Fast R-CNN	YOLO	SSD
Percentage	95	90	83	87

6.2.2 | Comparison of methods with different detection approaches

Precision and recall concepts are adopted to compare the performance of PADENet with that of other methods. The precision was defined as the number of true positives (TPs) divided by the sum of the TPs and false positives (FPs). The recall is represented by the number of TPs divided by the sum of TPs and false negatives (FNs), as shown in the following equation:

$$Precision = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN} \quad (5)$$

As demonstrated by the curves in Figure 11 , the PADENet approach produces higher quality results than the other approaches. These results suggest that the improvement in the multiscale prediction strategy results in better performance. Figure 12 presents the detection results of PADENet and the original Faster R-CNN. One of the reasons for the variations is that PADENet can obtain more useful information from small-scale defects. PADENet fuses the features and proposes potential defect areas at three different scales. The defect detection of other approaches degrades for relatively small training instances.

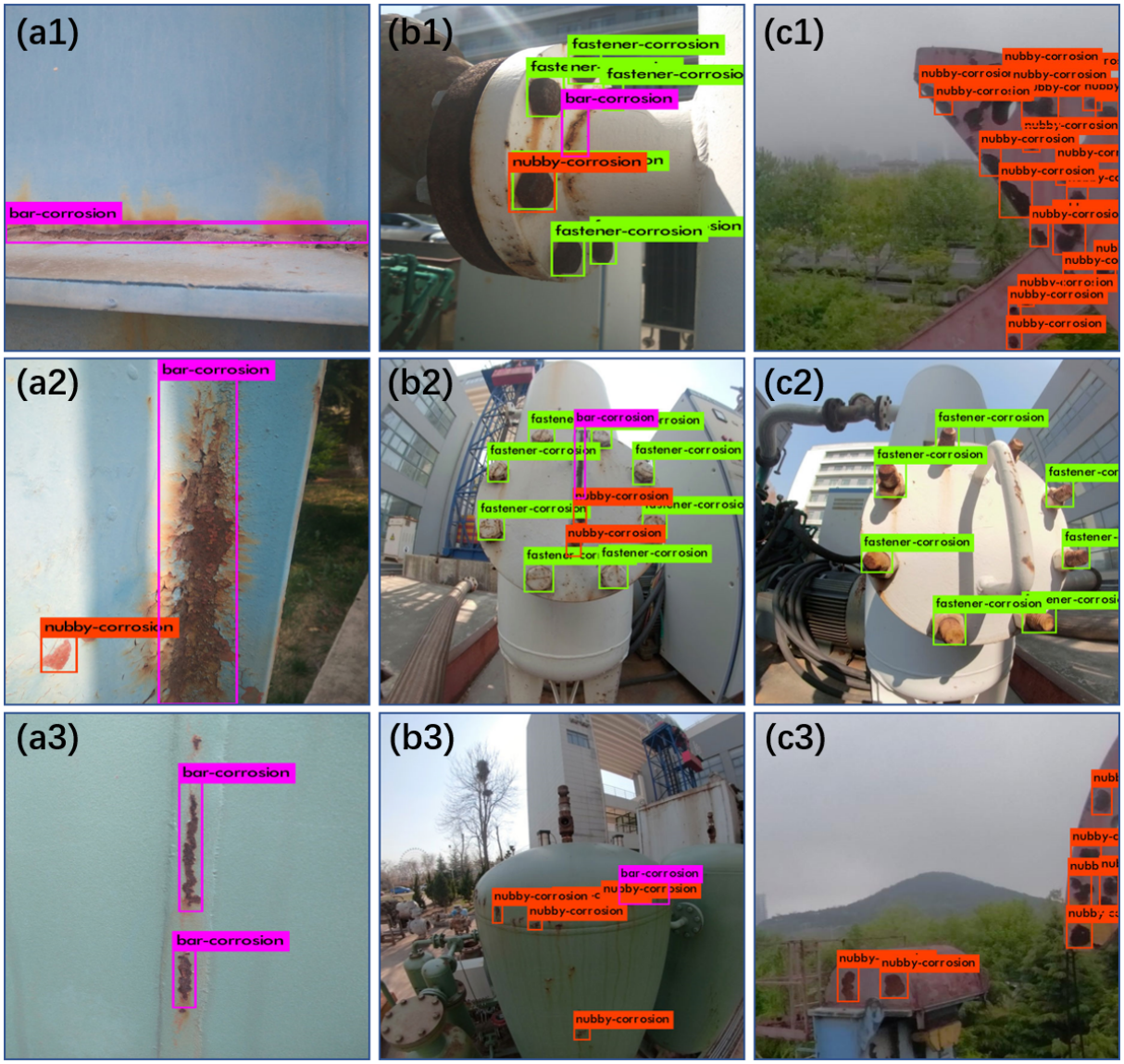

FIGURE 10 Sample images taken in different conditions with detection results and class labels. (a) Bar-corrosion detection. (b) Multitype defect detection. (c) Defects under various light conditions.

6.2.3 | Comparison of the method performance in different seasons

The performance can be further explored by examining the detection results in each season. The results show that defect features can be easily affected by the season and dynamic environmental illumination, as shown in Figure 13 .

The results illustrate that the detection performance decreases as the leaves change in autumn. The obtained results justify further development of the method for long-term detection requirements.

6.2.4 | Comparison of the methods using cross-validation assessment

To ensure the credibility of the results, the performance of the panoramic defect dataset must be determined by the 4-fold cross-validation method. The cross-validation assessment is repeated four times, with each of the four groups used only one time as the validation data, and the results are shown in Table 6 . The training images are randomly partitioned into four equal-sized groups according to the 4-fold principle.

In general, the PADENet method offers advantages over the other three detection methods, which can be explained in part

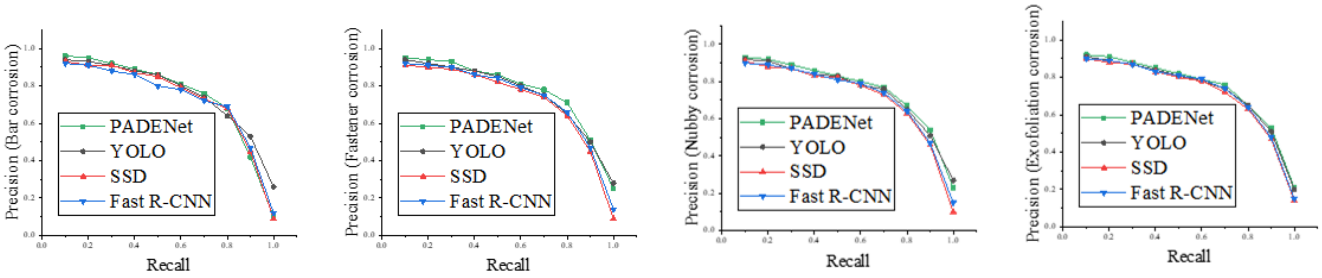

FIGURE 11 PR curves of the 4 detection methods. The experiments are conducted on a PC equipped with a Tesla V100 graphic processing unit (GPU) and an Intel Xeon Gold 5120 CPU @ 2.20 GHz. As the results demonstrate, the PR curve of PADENet is the outermost curve, showing that it has the best detection performance for steel surface damage detection.

TABLE 6 Performance of PADENet, Fast R-CNN, YOLO and SSD for the 4-fold validation assessment of the panoramic defect dataset.

Test fold	Approach	Average precision (%)			
		Exfoliation	Fastener corrosion	Nubby corrosion	Bar corrosion
1^{st} fold	PADENet	87.71	93.36	88.36	81.78
	Fast R-CNN	77.31	82.37	80.51	71.12
	YOLO	85.37	86.28	81.53	77.26
	SSD	81.21	87.11	81.03	73.11
2^{nd} fold	PADENet	88.64	92.38	87.41	78.89
	Fast R-CNN	78.63	84.27	83.21	68.57
	YOLO	84.25	85.81	83.12	76.82
	SSD	82.81	86.21	82.32	72.12
3^{rd} fold	PADENet	87.62	92.81	87.22	79.36
	Fast R-CNN	76.08	83.12	80.97	67.91
	YOLO	86.82	85.32	84.18	75.11
	SSD	81.13	87.41	80.71	71.39
4^{th} fold	PADENet	88.13	93.18	87.02	79.74
	Fast R-CNN	72.23	82.45	81.43	68.22
	YOLO	85.82	85.69	82.51	75.72
	SSD	80.32	86.13	81.45	72.43
Average	PADENet	88.03	92.93	87.50	79.94
	Fast R-CNN	76.06	83.05	81.53	68.96
	YOLO	85.57	85.78	82.84	76.23
	SSD	81.37	86.72	81.38	72.26

by the different results obtained for defects with different features in each category. The implementation of the panoramic defect dataset is unbiased in most cases (Kumar, 2008).

6.3 | Test images with compound challenges

To assess the performance of the trained PADENet, images with compound challenges are chosen for evaluation. The chosen images contain some ambient disturbances, such as trees, intense lights, shadows, fog, and unrelated buildings.

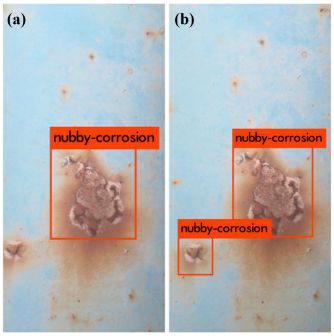

FIGURE 12 Detection comparison between Faster R-CNN and PADENet. (a) Small-scale defects cannot be detected by Faster R-CNN. (b) PADENet can successfully detect small-scale defects with the aid of the multiscale RPN.

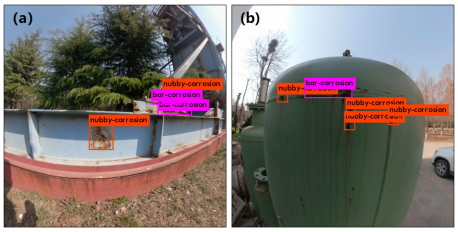

FIGURE 13 Comparison of detection in different seasons and under dynamic environmental illumination. (a) Defect detection results in spring/summer with a green leaf background. (b) Defect detection results in autumn/winter with a yellow/brown leaf background.

As shown in Figure 14 , the test results for the images with cluttered backgrounds are suboptimal. All the defects in Figure 14 are successfully detected and classified. The results show that all the damage is correctly detected under the disturbance of a background with similar clutter. The defects in Figure 14 a-b were partially in shadows. The defects in Figure 14 c have land with a similar color to corrosion as the background. All the defects in Figure 14 d were also detected and classified without the effect of unrelated buildings. All defects are successfully detected and classified.

6.4 | Demonstration of misdetected defects

Figure 15 presents detected faults and wrongly classified results. The steel defect in Figure 15 a was correctly detected as nubby corrosion but mislabeled as only one defect that is supposed to be three defects. The rust stain between defects may have caused PADENet to detect the three parts as an entire single defect. In Figure 15 b, there is one instance of fastener corrosion that is not detected. Since this image was taken on a sunny day, the nondetected fastener was covered by

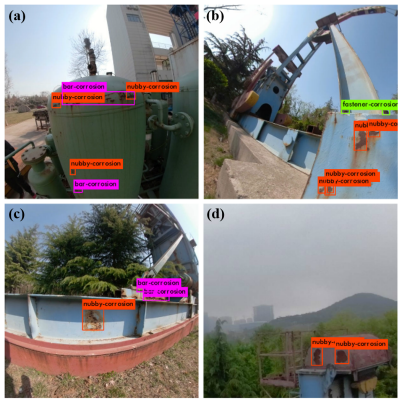

FIGURE 14 Images with compound challenges chosen for the evaluations. (a-b) Defects are partially in shadows. (c) Land with a similar color to corrosion. (d) Defects detected with unrelated buildings as background.

the shadow from the infrastructure. The PADENet still cannot distinguish the defect in low visibility. Figure 15 c shows another example that is misclassified by PADENet. The defects in Figure 15 c are a group of small nubby corrosions. There are many overlap areas between defects; therefore, it is difficult to label the border of each small defect clearly. We gave a unique challenge to PADENet in Figure 15 d. We chose special rust painting (QO1003, Q01004, Q01005, Q01006) from the J.N.Model Pigment series to mimic the steel defects. As the results presented in Figure 15d indicate, PADENet can successfully detect the real nubby corrosion. Due to the texture and color of model painting that are truly similar to the real rust, the fake corrosions are faultily labeled by PADENet.

Considering that only through the visible light reflected by the surface can the detection method tell the differences between real steel defects and fake defects with similar colors and textures, we plan to use hyperspectral sensors to overcome this challenge in future work.

6.5 | Further testing on newly captured panoramas

To further evaluate how well the trained PADENet detects infrastructural defects, we arrange an additional 5 new tests using novel 4992 × 2496 pixel panoramic images that have not been previously used in the testing procedure. These new images are taken by a UAV under various illumination situations.

Images that are not separated into subimages are evaluated. As seen in Figure 16 , the majority of defects are correctly detected, leaving only some small-scale damage unclassified. This result shows that the performance of PADENet is better than the performance of the original Faster R-CNN.

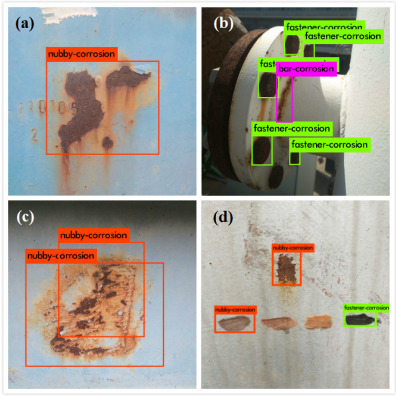

FIGURE 15 Demonstration of misdetected defects. (a) Defect mislabeled with only one damage. (b) One fastener corrosion not being detected. (c) A group of small defects is wrongly detected as two large groups. (d) Defects that mimic rust are faultily labeled.

FIGURE 16 An image not separated into subimages is evaluated by PADENet. The majority of defects are correctly detected, leaving only some small-scale damage unclassified.

In summary, PADENet successfully detects and classifies all the different sizes and types of surface damage. PADENet manages to perform effectively under the challenges of irregular topology and camera distortion.

7 | CONCLUSIONS

A surface damage detection method is proposed for detecting four types of defects in 360 ° panoramas: bar corrosion, nubby corrosion, fastener corrosion, and exfoliation. The panoramic images required for the training and testing are taken by a UAV equipped with a dual fisheye camera. The raw images are horizontally flipped for data augmentation. Then, a total of 16,000 images are divided into 12,800 images for training and 3,200 images for testing. The robustness of the trained PADENet is evaluated on 5 extra-large images with a resolution of 4992×2496 pixels that are not used for the testing procedure. The performance of the trained PADENet is further compared with that of the Fast R-CNN, YOLO and SSD methods. We demonstrate that the AP ratings for the four damage types (bar corrosion, fastener corrosion, nubby corrosion, and exfoliation) are 80.78, 92.41, 87.81 and 88.36, respectively. In addition, the 4-fold cross-validation method is used to assess the panoramic dataset. The comparative study shows that the proposed PADENet can provide optimal results, and large bias errors do not occur in the panoramic dataset.

In general, the fact that the Faster R-CNN-based method can detect damage features from a large number of training images to train a reliable classifier is a very large advantage. However, this feature also limits the implementation of the Faster R-CNN-based method due to the requirement of vast amounts of training data.

In future detection tests, more samples with various kinds of surface damage under different challenges will be captured and added to the dataset. Additionally, more kinds of UAV platforms with longer flight times and wider coverage ranges need to be tested to improve the usage of autonomous systems to replace human-oriented visual inspection assessments.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61701541, Project U1906217, Grant 51975586 and Grant 61971444 and supported in part by the Fundamental Research Funds for the Central Universities under Grant 19CX02021A. The authors would like to thank the technicians from the Oil Industry Training Center of the China University of Petroleum for the rust painting and flight testing support.

References

Abdel-Qader, I., Abudayyeh, O., & Kelly, M. E. (2003). Analysis of edge-detection techniques for crack identification in bridges. *Journal of Computing in Civil Engineering*, *17*(4), 255–263.

Adeli, H., & Jiang, X. (2008). *Intelligent infrastructure: neural networks, wavelets, and chaos theory for intelligent transportation systems and smart structures*. CRC press.

Agnisarman, S., Lopes, S., Madathil, K. C., Piratla, K., & Gramopadhye, A. (2019). A survey of automation-enabled human-in-the-loop systems for infrastructure visual inspection. *Automation in Construction*, *97*, 52–76.

Ahmadlou, M., & Adeli, H. (2010). Enhanced probabilistic neural network with local decision circles: A robust classifier. *Integrated Computer-Aided Engineering*, *17*(3), 197–210.

Alam, K. M. R., Siddique, N., & Adeli, H. (2020). A dynamic ensemble learning algorithm for neural networks. *Neural Computing and Applications*, *32*(12), 8675–8690.

Amezquita-Sanchez, J. P., & Adeli, H. (2015). Synchrosqueezed wavelet transform-fractality model for locating, detecting, and quantifying damage in smart highrise building structures. *Smart Materials and Structures*, *24*(6), 065034.

Bang, S., Park, S., Kim, H., & Kim, H. (2019). Encoder–decoder network for pixel-level road crack detection in black-box images. *Computer-Aided Civil and Infrastructure Engineering*.

Bertozzi, M., Castangia, L., Cattani, S., Prioletti, A., & Versari, P. (2015). 360 detection and tracking algorithm of both pedestrian and vehicle using fisheye images. In *2015 ieee intelligent vehicles symposium (iv)* (pp. 132–137).

Cha, Y.-J., Choi, W., Suh, G., Mahmoudkhani, S., & Büyüköztürk, O. (2018). Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil and Infrastructure Engineering*, *33*(9), 731–747.

Chen, S.-H., & Perng, D.-B. (2016). Automatic optical inspection system for ic molding surface. *Journal of Intelligent Manufacturing*, *27*(5), 915–926.

Cho, C.-S., Chung, B.-M., & Park, M.-J. (2005). Development of real-time vision-based fabric inspection system. *IEEE Transactions on Industrial Electronics*, *52*(4), 1073–1079.

Chun, P.-j., Hashimoto, K., Kataoka, N., Kuramoto, N., & Ohga, M. (2015). Asphalt pavement crack detection using image processing and naive bayes based machine learning approach. *Journal of Japan Society of Civil Engineers, Ser. E1 (Pavement Engineering)*, *70*(3).

Deng, F., Zhu, X., & Ren, J. (2017). Object detection on panoramic images based on deep learning. In *2017 3rd international conference on control, automation and robotics (iccar)* (pp. 375–380).

Dueholm, J. V., Kristoffersen, M. S., Satzoda, R. K., Moeslund, T. B., & Trivedi, M. M. (2016). Trajectories and maneuvers of surrounding vehicles with panoramic camera arrays. *IEEE Transactions on Intelligent Vehicles*, *1*(2), 203–214.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, *88*(2), 303–338.

Fan, R., Jiao, J., Pan, J., Huang, H., Shen, S., & Liu, M. (2019). Real-time dense stereo embedded in a uav for road inspection. In *Proceedings of the ieee conference on computer vision and pattern recognition workshops* (pp. 0–0).

Gao, Y., & Mosalam, K. M. (2018). Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, *33*(9), 748–768.

German, S., Brilakis, I., & DesRoches, R. (2012). Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments. *Advanced Engineering Informatics*, *26*(4), 846–858.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the ieee international conference on computer vision* (pp. 1440–1448).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).

Huang, S.-H., & Pan, Y.-C. (2015). Automated visual inspection in the semiconductor industry: A survey. *Computers in industry*, *66*, 1–10.

Jang, K., An, Y.-K., Kim, B., & Cho, S. (2020). Automated crack evaluation of a high-rise bridge pier using a ring-type climbing robot. *Computer-Aided Civil and Infrastructure Engineering*.

Jiang, X., & Adeli, H. (2010). Pseudospectra, music, and dynamic wavelet neural network for damage detection of highrise building. *International Journal for Numerical Methods in Engineering*, *71*(5), 606-629.

Kang, D., & Cha, Y.-J. (2018). Autonomous uavs for structural health monitoring using deep learning and an ultrasonic beacon system with geo-tagging. *Computer-Aided Civil and Infrastructure Engineering*, *33*(10), 885–902.

Koch, C., & Brilakis, I. (2011). Pothole detection in asphalt pavement images. *Advanced Engineering Informatics*, *25*(3), 507–515.

Koch, C., Paal, S. G., Rashidi, A., Zhu, Z., König, M., & Brilakis, I. (2014). Achievements and challenges in machine vision-based inspection of large concrete structures. *Advances in Structural Engineering*, *17*(3), 303–318.

Koch, C., Zhu, Z., Paal, S. G., & Brilakis, I. (2015). Machine vision techniques for condition assessment of civil infrastructure. In *Integrated imaging and vision techniques for industrial inspection* (pp. 351–375). Springer.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

Kumar, A. (2008). Computer-vision-based fabric defect detection: A survey. *IEEE transactions on industrial electronics*, *55*(1), 348–363.

Lenjani, A., Yeum, C. M., Dyke, S., & Bilionis, I. (2020). Automated building image extraction from 360° panoramas for postdisaster evaluation. *Computer-Aided Civil and Infrastructure Engineering*, *35*(3), 241–257.

Li, R., Yuan, Y., Zhang, W., & Yuan, Y. (2018). Unified vision-based methodology for simultaneous concrete

defect detection and geolocalization. *Computer-Aided Civil and Infrastructure Engineering*, *33*(7), 527–544.

Li, S., Zhao, X., & Zhou, G. (2019). Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network. *Computer-aided Civil and Infrastructure Engineering*, *34*(7), 616–634.

Li, Y., Chen, Y., Wang, N., & Zhang, Z. (2019). Scale-aware trident networks for object detection. In *Proceedings of the ieee international conference on computer vision* (pp. 6054–6063).

Li, Y., Tong, G., Gao, H., Wang, Y., Zhang, L., & Chen, H. (2019). Pano-rsod: A dataset and benchmark for panoramic road scene object detection. *Electronics*, *8*(3), 329.

Liang, X. (2019). Image-based post-disaster inspection of reinforced concrete bridge systems using deep learning with bayesian optimization. *Computer-Aided Civil and Infrastructure Engineering*, *34*(5), 415–430.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21–37).

Liu, Y.-F., Nie, X., Fan, J.-S., & Liu, X.-G. (2019). Image-based crack assessment of bridge piers using unmanned aerial vehicles and three-dimensional scene reconstruction. *Computer-Aided Civil and Infrastructure Engineering*.

Morgenthal, G., & Hallermann, N. (2014). Quality assessment of unmanned aerial vehicle (uav) based visual inspection of structures. *Advances in Structural Engineering*, *17*(3), 289–302.

Narazaki, Y., Hoskere, V., Hoang, T. A., Fujino, Y., Sakurai, A., & Spencer Jr, B. F. (2020). Vision-based automated bridge component recognition with high-level scene consistency. *Computer-Aided Civil and Infrastructure Engineering*, *35*(5), 465–482.

Ni, F., Zhang, J., & Chen, Z. (2019). Pixel-level crack delineation in images with convolutional feature fusion. *Structural Control and Health Monitoring*, *26*(1), e2286.

Pereira, D. R., Piteri, M. A., Souza, A. N., Papa, J. P., & Adeli, H. (2020). Fema: a finite element machine for fast learning. *Neural Computing and Applications*, *32*(10), 6393–6404.

Rafiei, M. H., & Adeli, H. (2017a). A new neural dynamic classification algorithm. *IEEE transactions on neural networks and learning systems*, *28*(12), 3074–3083.

Rafiei, M. H., & Adeli, H. (2017b). A novel machine learning-based algorithm to detect damage in high-rise building structures. *The Structural Design of Tall and Special Buildings*, *26*(18), e1400.

Rafiei, M. H., & Adeli, H. (2018). A novel unsupervised deep learning model for global and local health condition assessment of structures. *Engineering Structures*, *156*, 598–607.

Rafiei, M. H., Khushefati, W. H., Demirboga, R., & Adeli, H. (2017). Supervised deep restricted boltzmann machine for estimation of concrete. *ACI Materials Journal*, *114*(2).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

Sajedi, S. O., & Liang, X. (2020). Uncertainty-assisted deep vision structural health monitoring. *Computer-Aided Civil and Infrastructure Engineering*.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Uijlings, J. R., Van De Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, *104*(2), 154–171.

Vaghefi, K., Oats, R. C., Harris, D. K., Ahlborn, T. T. M., Brooks, C. N., Endsley, K. A., … Dobson, R. (2011). Evaluation of commercially available remote sensors for highway bridge condition assessment. *Journal of Bridge Engineering*, *17*(6), 886–895.

Yang, W., Qian, Y., Kämäräinen, J.-K., Cricri, F., & Fan, L. (2018). Object detection in equirectangular panorama. In *2018 24th international conference on pattern recognition (icpr)* (pp. 2190–2195).

Yang, W., Qian, Y., Kämäräinen, J., Cricri, F., & Fan, L. (2018). Object detection in equirectangular panorama. In *2018 24th international conference on pattern recognition (icpr)* (p. 2190-2195).

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818–833).

Zhang, Y., Xiao, X., & Yang, X. (2017). Real-time object detection for 360-degree panoramic image using cnn. In *2017 international conference on virtual reality and visualization (icvrv)* (pp. 18–23).

Zhou, G., Li, C., & Cheng, P. (2005). Unmanned aerial vehicle (UAV) real-time video registration for forest fire monitoring. In (Vol. 3, p. 1803-06).

How to cite this article: Cai L, Leijian Y, Jiaxing Y, Zhongwei L, Peng R, Xiao B, Yonghong L. Autonomous detection of multiple types of steel defects from 360° panoramas using deep neural networks. *Comput Aided Civ Inf.* , 2020;00:1–6.