Deep learning–based image instance segmentation for moisture marks of shield tunnel lining

Shuai Zhao, Dong Ming Zhang*, Hong Wei Huang*

Key Laboratory of Geotechnical and Underground Engineering, Department of Geotechnical Engineering, Tongji University, Siping Road, 1239 Shanghai, China

ABSTRACT

This paper presents a method for the image instance segmentation of the moisture marks of shield tunnel lining using a mask-region-based convolutional neural network (Mask R-CNN) algorithm. The authors' previously proposed fully convolutional network (FCN) framework and the moisture-mark detection framework have been combined into a unified Mask R-CNN framework. A total of 5031 images covering five scales were collected and annotated to train this deep-learning (DL)-based algorithm to identify the moisture marks in images. Three steps are detailed for instance segmentation: feature extraction, the generation of region proposals, and moisture-mark identification. A high-quality segmentation mask for the moisture marks is generated, and the moisture-mark area is obtained by counting the pixels with a value of 1 in the polygon generated for moisture marks during the test process of the trained Mask R-CNN model. The proposed method is validated by an experimental study, and the results are compared with those obtained by the authors' previous FCN method and two conventional methods—the region growing algorithm (RGA) and Otsu algorithm (OA). The accuracy, F_1 score, and intersection over union (IoU) for the proposed method are superior than those for the FCN, RGA, and OA with respect to 503 test images. The inference time for the proposed method is considerably shorter than that for the FCN and RGA and slightly longer than that for the OA.

1. Introduction

A metro shield tunnel is commonly used to relieve the traffic pressure on the ground. When a metro shield tunnel degrades owing to ageing or environmental disturbances, various defects such as moisture marks will develop and deteriorate during the service time. Water leakage is considered the most significant structural issue of the metro tunnels in Shanghai, China (Wang et al., 2011). The failure to detect water leakage in advance caused the large deformation of a metro tunnel in Shanghai in 2008 (Huang et al., 2017a). Moisture marks are formed by water ingress through the segmental joints or cracks in concrete. Once the intrusive water exceeds a certain limit of volume or area, it may serve as a warning of the serious deterioration of a tunnel (Dawood et al., 2018). Therefore, to ensure safety, human-based onsite inspection is regularly conducted at midnight over a limited duration (Ai et al., 2016; Huang et al., 2017b). The accuracy and precision of inspection by the naked eye significantly depend on the trained inspectors and inevitably bias the evaluation of the tunnel's conditions. Owing to the disadvantages of human-conducted onsite inspections, computer-vision-based methods have been used to detect moisture marks. Ukai and Nagamine (2011) used an image smoothing technique and dynamic threshold value processing to extract the moisture marks in images. The detection algorithm focused on the pixel grey values. However, objects present on the tunnel lining, such as segmental joints and pipes, generally have similar pixel grey values. Thus, these distractors were easily incorrectly identified as moisture marks.

Dawood et al. (2018) used image pre-processing techniques and an artificial neural network (ANN) to detect and quantify the moisture marks in an image, respectively. The detection and quantification results could satisfy the needs of engineering inspection. Similarly, Hu et al. (2010) used an ANN to distinguish moisture marks from the background of an image. A satisfactory detection result was obtained; however, the performance will be significantly reduced when the background and light reflection are more complex.

Terrestrial laser scanning (TLS) and mobile laser scanning (MLS) have been used for tunnel inspection owing to their good performance for archiving intensive point cloud data. Tan et al. (2016), Xu et al. (2018), and Wu and Huang (2018) have used corrected intensive point data obtained by TLS or MLS to detect moisture marks and have combined three-dimensional (3D) point clouds to quantify moisture marks. Yu et al. (2018) proposed a method that integrates laser scanning and infrared thermal imaging to diagnose moisture marks.

Computer-vision-based methods will produce a large amount of image data including an image with defects and an image without defects. Therefore, the efficient detection of defects from the large image database has become a new challenge for engineers. Previous studies on defect detection are based on image processing techniques (IPTs) (Abdel-Qader et al., 2003; Nishikawa et al., 2012; German et al., 2012). These IPTs, however, are difficult to adapt to the complex environment of tunnels in service, e.g. the changes in lighting and shadows in dark tunnels.

Deep learning (DL) has received a considerable amount of attention in recent years. It discovers the intricate structure of large datasets by using a backpropagation algorithm (LeCun et al., 2015). Convolutional neural networks (CNNs), a DL-based architecture, are designed to process data that come in the form of multiple arrays, such as images. There have been numerous applications of CNNs in civil engineering for the detection of structural defects (Soukup and Huber-Mörk, 2014; Cha et al., 2017a), and they have proven to be very effective.

Recently, based on a deep CNN framework, models have been developed and have rapidly improved object detection and semantic segmentation results. These DL models have been applied to civil engineering. Cha et al. (2017b) modified and trained a faster region-based CNN (Faster R-CNN) model (Ren et al., 2017) to detect five types of surface damage (concrete cracks, medium- and high-level steel corrosion, bolt corrosion, and steel delamination) and achieved good results. Xue and Li (2018) adopted a fully convolutional network (FCN) and region-based FCN (R-FCN) (Dai et al., 2016) to classify and detect defects (crack and moisture marks), respectively. The above studies focused on defect detection; however, the sizes of the defects, such as the area of moisture marks, were not quantified. More specifically, the sizes of defects are difficult to obtain solely on the basis of the above DL-based methods. Huang et al. (2018) employed an FCN (Long et al., 2015) for the semantic segmentation of cracks and moisture marks. However, the location of a defect on the tunnel lining usually is difficult to determine owing to the background of the image. Additionally, the areas of different moisture marks in one image are difficult to obtain. Hence, the goal of our work is to build an instance segmentation framework to overcome the above obstacles.

Instance segmentation combines two main tasks: object detection and semantic segmentation. The first task is to classify individual objects and localise each object using a bounding box, and the second task is to classify each pixel into a fixed set of categories without differentiating object instances (Fig. 1). A mask-region-based convolutional neural network (Mask R-CNN) is a recently developed DL algorithm that can deal with the instance segmentation task (He et al., 2017). This method efficiently detects the objects in an image while simultaneously generating a high-quality segmentation mask for each instance. Mask R-CNNs have been used in some fields for the delineation and classification of ice-wedge polygons (Zhang et al., 2018), environmental monitoring and exploration (Zurowietz et al., 2018), and the recovery of a droplet's 3D shape (Lu et al., 2018) and have proven to be very effective.

Because the aim of our study is image instance segmentation for moisture marks, a Mask R-CNN is adopted and trained to fulfil the requirements of the specific environments of the tunnels in service. Before a detailed introduction of the Mask R-CNN used in this study, an image dataset containing 5031 images for the instance segmentation of moisture marks is first discussed. The image dataset is enlarged using images containing moisture marks annotated with LabelMe (Wada, 2016). After establishing image datasets, three steps—feature extraction, the generation of region proposals, and moisture-mark identification—are introduced in detail to explain the Mask R-CNN model. The polygon that is output during the test process is used to generate a mask and calculate the area of the moisture marks. The accuracy, F_1 score, intersection over union (IoU), and inference time of the proposed Mask R-CNN are compared to those of an FCN, the region growing algorithm (RGA), and the Otsu algorithm (OA) with respect to the test images. Then, the reasons why different results were obtained by these four methods are briefly discussed. Finally, the concluding remarks are mentioned at the end of this paper.

2. Dataset of tunnel lining defects

The detection and segmentation tasks strongly rely on large amounts of labelled image data and computing power. Previously, a significant issue associated with the use of a CNN was the requirement of a vast amount of labelled image data and the associated high computational cost, but this issue has been overcome through the development of labelling techniques and parallel computation using graphics processing units (GPUs). Nevertheless, there are few well-annotated open-source datasets that contain information about tunnel moisture marks at present. Therefore, it is crucial to collect a sufficient number of images containing moisture marks to achieve the image instance segmentation of moisture marks.

2.1. Image acquisition of the metro shield tunnel lining

Metro tunnel inspection is often conducted at midnight and must be completed within 2–3 h in order to guarantee the daily operation of the metro tunnel. Therefore, it is difficult to collect large numbers of images of the tunnel lining surface in a short duration. As a result, images must be acquired with a high efficiency and precision. However, existing tunnel inspection equipment cannot meet these requirements. The Moving Tunnel Inspection (MTI-200a) equipment previously developed by the authors can meet the requirements of metro tunnel inspection and image acquisition. This MTI system comprises six high-resolution linear charge coupled device (CCD) cameras, 12 light-emitting diodes (LEDs), a computer, a monitor, an encoder, and a battery, as shown in Fig. 2. The MTI system is relatively light and easy to move. A captured image has a high quality and can be cropped and labelled directly. The details of this equipment can be found elsewhere (Huang et al., 2017c,
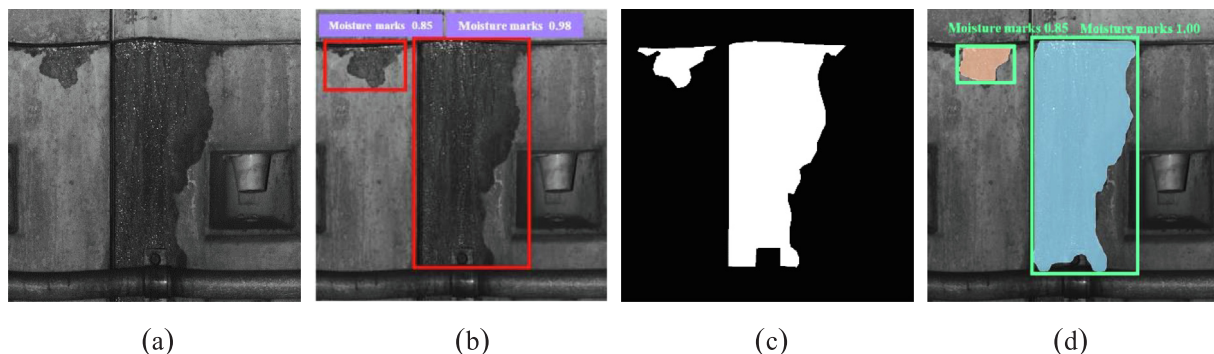

(a)　　　　　　(b)　　　　　　(c)　　　　　　(d)

Fig. 1. Process of image recognition: (a) raw image, (b) object (moisture mark) detection, (c) semantic segmentation, and (d) instance segmentation.
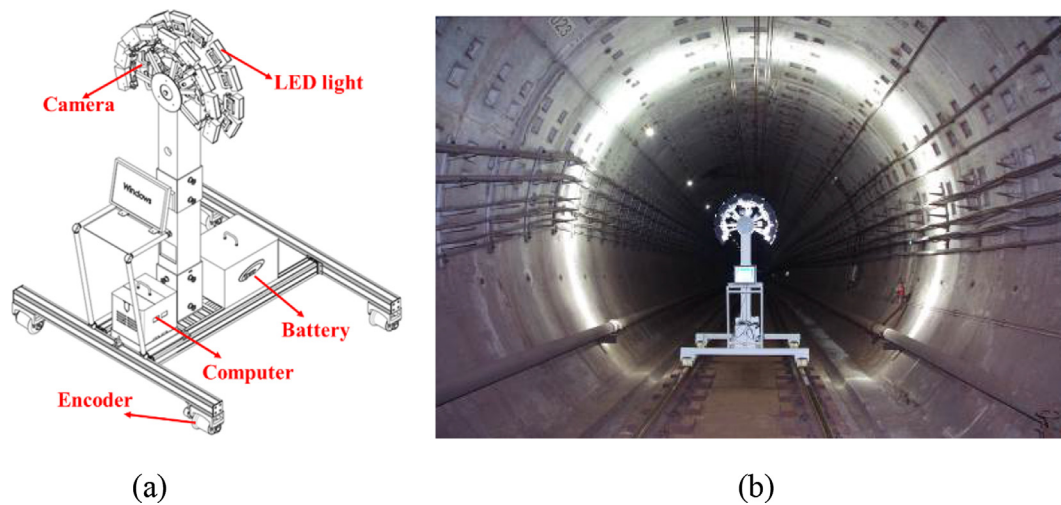
Fig. 2. (a) Schematic of MTI-200a and (b) photograph of MTI-200a in a tunnel.

2018).

2.2. Establishment of image datasets

Inspection tasks were carried out in Shanghai metro lines 1, 2, 4, 7, 8, 10, and 12. The captured images were stored with a resolution of 1,000 × 7,448 pixels within each camera. Images with a resolution of 3,000 × 7,448 pixels were obtained by stitching the captured images. Then, the images were cropped to a resolution of 3,000 × 3,724 pixels, and the images containing moisture marks were selected. To ensure that the size of an image corresponds to the size of the moisture marks, the selected images were further cropped to five scales with different resolutions: 800 × 800, 1200 × 1200, 1600 × 1600, 2000 × 2000, and 2400 × 2400 pixels. Finally, all selected images were annotated by LabelMe. The moisture marks were annotated along boundaries by drawing a polygon. When the polygon was completely drawn, a label was written into the dialogues, as shown in Fig. 3. Consequently, an annotation file containing the moisture marks' width, length, and other information was generated, and the annotation file was converted into a format similar to that for the Microsoft COCO dataset (Lin et al., 2015) through Python. Fig. 4 shows part of the annotated images.

The dataset containing annotated images is named the Tunnel Lining Moisture Marks in Context (TLWLCO) dataset. The format of the annotation file of TLWLCO dataset is similar to that of the Microsoft COCO dataset. The TLWLCO dataset contains 5,031 annotated images covering five scales of resolution: 800 × 800, 1200 × 1200,

Fig. 3. Moisture marks annotated by LabelMe.

1600 × 1600, 2000 × 2000, and 2400 × 2400 pixels. It was divided into training, validation, and testing datasets. Following a study by Shahin et al. (2004), 90% of the data (i.e. 4,528 images) were used for calibration, and 10% of the data (i.e. 503 images) were used for testing. The calibration data were further divided into 80% for training (i.e. 3,622 images) and 20% for validation (i.e. 906 images). The Mask R-CNN will automatically perform horizontal flipping (Fig. 5) of the training and validation sets for data augmentation.

2.3. Unique properties of the moisture marks

Moisture marks possess unique properties including the colour, texture, and edge information (Dawood et al., 2018). Such properties differentiate moisture marks from other concrete surface defects. The colour and edge information are the two important indicators for object recognition. Moisture marks are darker than other objects owing to the low reflectance, where the edges of the moisture marks have larger gradient grey values. In addition, the texture is another crucial indicator. Moisture marks have a smooth and fine texture, unlike other surface defects such as spalling, which displays a coarse texture in an image. It is these three unique properties that motivate the use of Mask R-CNN segmentation algorithm. This algorithm allows a machine or computer to be fed with raw image data and to automatically discover the representations needed for moisture-mark detection. An image comes in the form of an array of pixel values. Given the pixels, the convolutional layer of the Mask R-CNN can easily identify the edges of moisture marks by comparing the brightness values of neighbouring pixels. The next convolutional layers detect motifs by spotting the particular arrangements of edges and the texture of the moisture marks. The subsequent convolutional layers assemble motifs into larger combinations that correspond to parts of the moisture marks. Finally, this description of an image in terms of the moisture-mark part can be used to recognise the moisture marks in the image. The key of the Mask R-CNN model is that these layers of features of moisture marks are learned from data using a self-learning procedure. Thus, a well-trained Mask R-CNN model can segment moisture marks from the background of an image, regardless of the distractors such as segmental joints, pipes, and bolt holes.

3. Framework for the image instance segmentation of moisture marks

To localise moisture marks in images while simultaneously generating high-quality segmentation masks for them, the Mask R-CNN is modified and used. The original Mask R-CNN consists of a Faster R-CNN
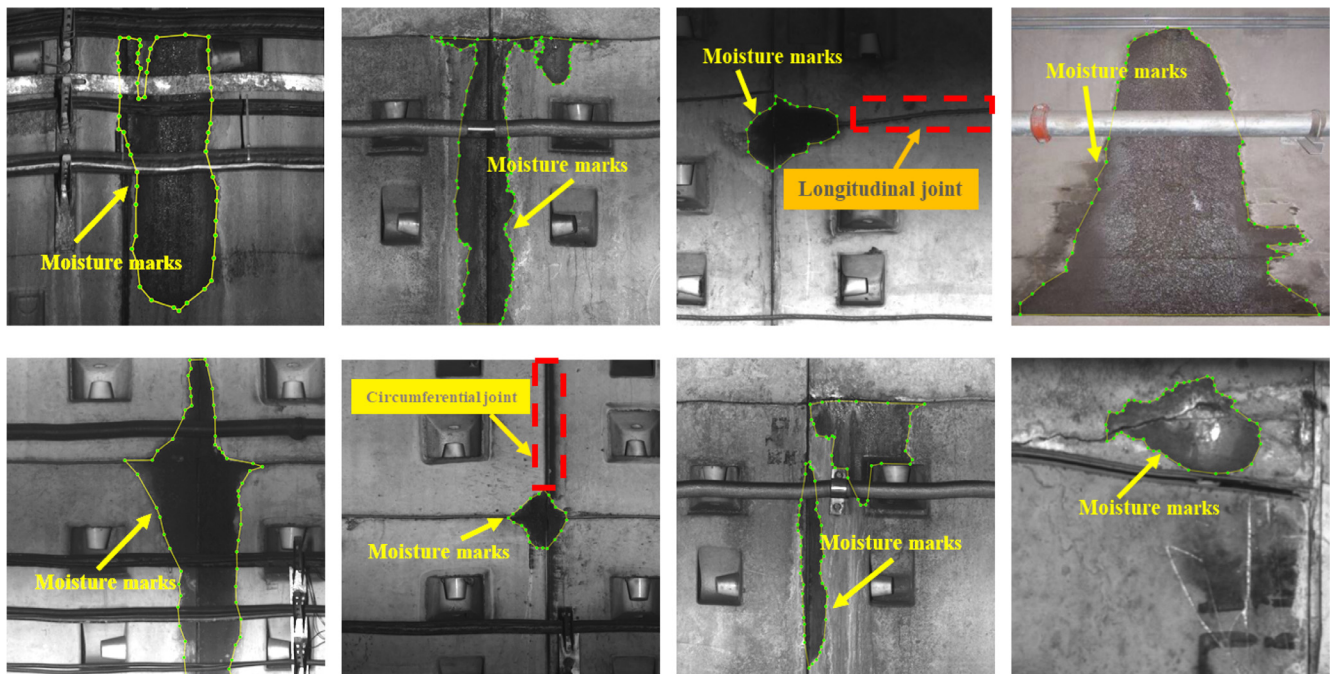
Fig. 4. Annotated images.

for classification and bounding-box regression and an FCN for predicting segmentation masks. The details of the Mask R-CNN are explained in this section. For clarity, the three steps are (i) the extraction of the features of moisture marks over an entire image with a convolutional backbone architecture, (ii) the generation of proposals of moisture marks with a region proposal network (RPN), and (iii) the classification, bounding-box recognition, and mask prediction of moisture marks with a head architecture.

3.1. Step 1: extraction of the features of moisture marks with a backbone architecture

A deep CNN is often used as a backbone architecture to compute the feature hierarchies of moisture marks layer-by-layer. The CNN's feature hierarchy produces feature maps with different spatial resolutions but introduces large semantic gaps caused by the different depths of the CNN's convolutional layer. High-resolution maps have low-level features, whereas low-resolution maps have high-level features. Recent research on moisture-mark recognition (Huang et al., 2018; Xue and Li, 2018; Cha et al., 2017a, 2017b) used CNNs to produce a single high-level coarser-resolution feature map from which the moisture marks were predicted (Fig. 6(a)). Thus, it misses the opportunity to re-use the higher-resolution maps of the feature hierarchy. However, for some small moisture marks, the semantic features may be lost when the CNN

executes pooling to the last layer. Therefore, the higher-resolution maps with low-level features are important for recognising small objects such as small moisture marks.

Feature pyramid networks (FPNs) (Lin et al., 2017) are a solution to the aforementioned problem. They leverage the pyramidal shape of a CNN's feature hierarchy while creating a feature pyramid that has strong semantics at all scales without sacrificing the representational power, speed, or memory. Therefore, an FPN is used as the backbone architecture of the Mask R-CNN in this study to extract the features of moisture marks in images. An FPN combines low-resolution semantically strong features with high-resolution semantically weak features via a top-down pathway and lateral connections (Figs. 6(b) & 7). As a result, a feature pyramid that has rich semantics at all levels is quickly built from a single input image scale.

Fig. 7 shows the structure of an FPN. A 1×1 convolutional layer is attached to conv5_x layer. After convolution by it, the coarsest resolution feature map is produced, which has the semantically strong features of moisture marks. Then, the spatial resolution is up-sampled by a factor of two. The up-sampled map is then merged with the corresponding bottom-up map (which passes through a 1×1 convolutional layer to reduce the channel dimensions) by element-wise addition. This process is implemented until the finest resolution map is generated. Finally, a 3×3 convolutional layer is appended to each merged map to generate the final feature map in order to reduce the aliasing effect of
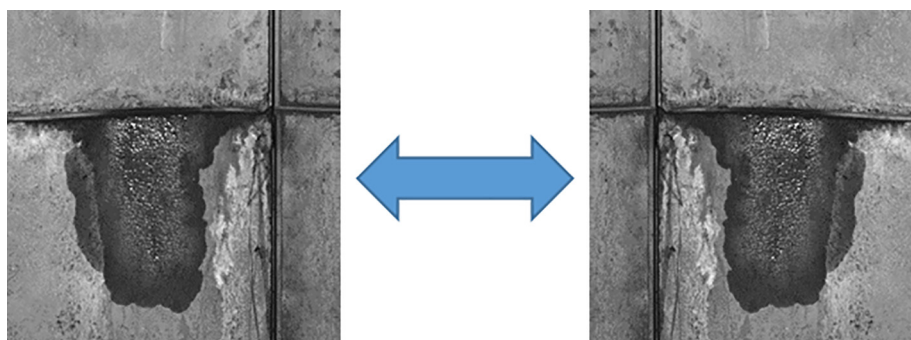

Fig. 5. Raw image (left) and horizontally flipped image (right) for data augmentation.

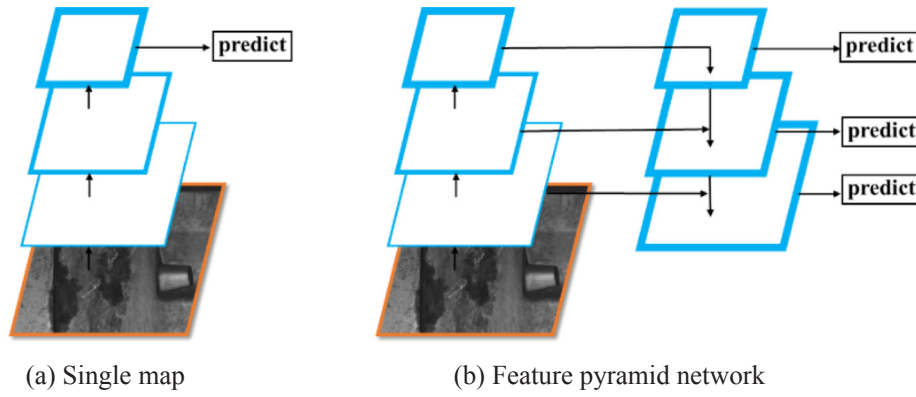(a) Single map   (b) Feature pyramid network

Fig. 6. Feature pyramid. Thicker blue outlines denote semantically stronger features. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
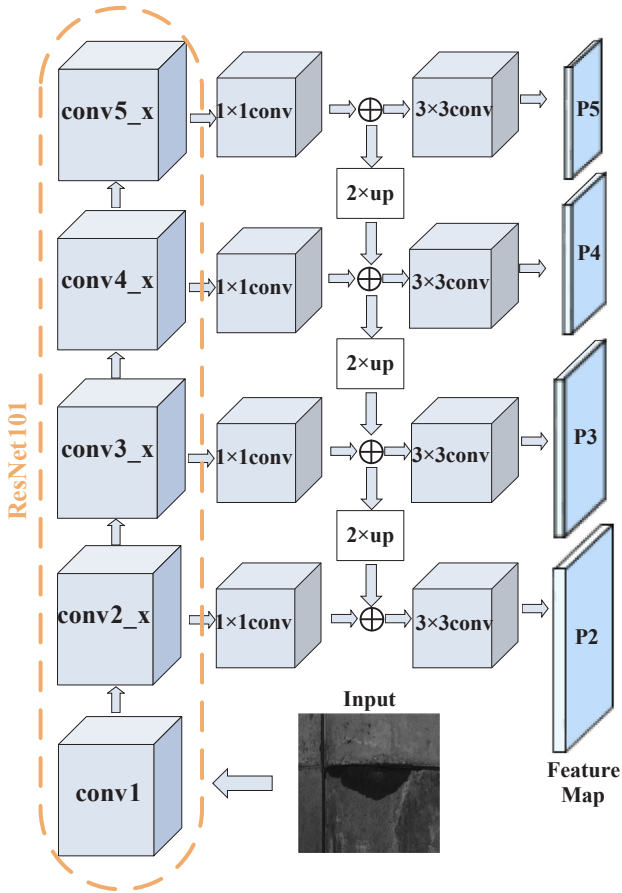

Fig. 7. Structure of an FPN.

up-sampling. The final generated set of feature maps is called {P2, P3, P4, P5}, where moisture-mark prediction are independently carried out at each level (Fig. 6(b)).

3.2. Step 2: generation of region proposals for moisture marks with a region proposal network

Object detection networks depend on region proposal algorithms to hypothesise object locations, and the computation of region proposals is a bottleneck for the running time of detection networks. An RPN solves this problem and enables nearly cost-free region proposals by sharing features with the backbone architecture. An RPN is constructed via a small subnetwork (FCN). The small subnetwork is evaluated with dense

3×3 sliding windows on top of a single-scale convolutional feature map output by the backbone architecture, performing object/non-object binary classification and bounding-box regression. This is realised by a 3×3 convolutional layer, followed by two sibling 1×1 convolutions for classification and regression (Fig. 8). At each sliding-window location, anchors with multiple pre-defined scales (three scales and three aspect ratios {1:2, 1:1, 2:1}) are simultaneously predicted (Fig. 8). These anchors are used as classification and regression references.

In this study, an FPN is used to compute the features of moisture marks in images and produce feature maps with four scales {P2, P3, P4, P5}; therefore, it is not necessary to have multi-scale anchors on a specific level. The anchors of the aspect ratios {1:2, 1:1, 2:1} are also used at each level. Thus, there are 12 anchors in total over the pyramid (Fig. 9).

At the training stage, a total of 2,000 proposals per FPN level are first selected. However, these 2,000 RPN proposals highly overlap each other. To reduce redundancy, non-maximum suppression (NMS) (Ren et al., 2017) is adopted for the proposal regions based on their class scores. Finally, the top-256 ranked proposals are selected for classification, bounding-box regression, and segmentation.

3.3. Step 3: moisture-mark identification with the head architecture

Fig. 10 shows the structure of the Mask R-CNN including the backbone architecture, RPN, and head architecture. A backbone architecture (FPN) takes an entire image containing moisture marks as its input. The FPN first processes the whole image with several convolutional and max pooling layers to produce feature maps with four scales. Then, the RPN uses these feature maps to produce a set of moisture-mark proposals. With the cross-boundary anchors ignored and NMS adopted, the top-256 ranked moisture-mark proposals based on their class scores are selected for the region of interest (RoI) align layer (He et al., 2017). Then, for each moisture-mark proposal, an RoI align layer extracts a fixed-length feature vector from the feature map. Each feature vector is fed into a sequence of fully connected layers for classification and bounding-box regression by the head architecture. The mask branch (FCN) simultaneously generates a high-quality segmentation mask for each RoI in parallel with the existing branch for classification and bounding-box regression.

During training, the multi-task loss is defined for each sampled RoI as $L = L_{cls} + L_{box} + L_{mask}$. The classification loss is

$$L_{cls}(p, u) = -\log p_u \tag{1}$$

where p_u is the predicted probability of class u being an object and p is computed by a softmax function over the outputs of a fully connected layer.

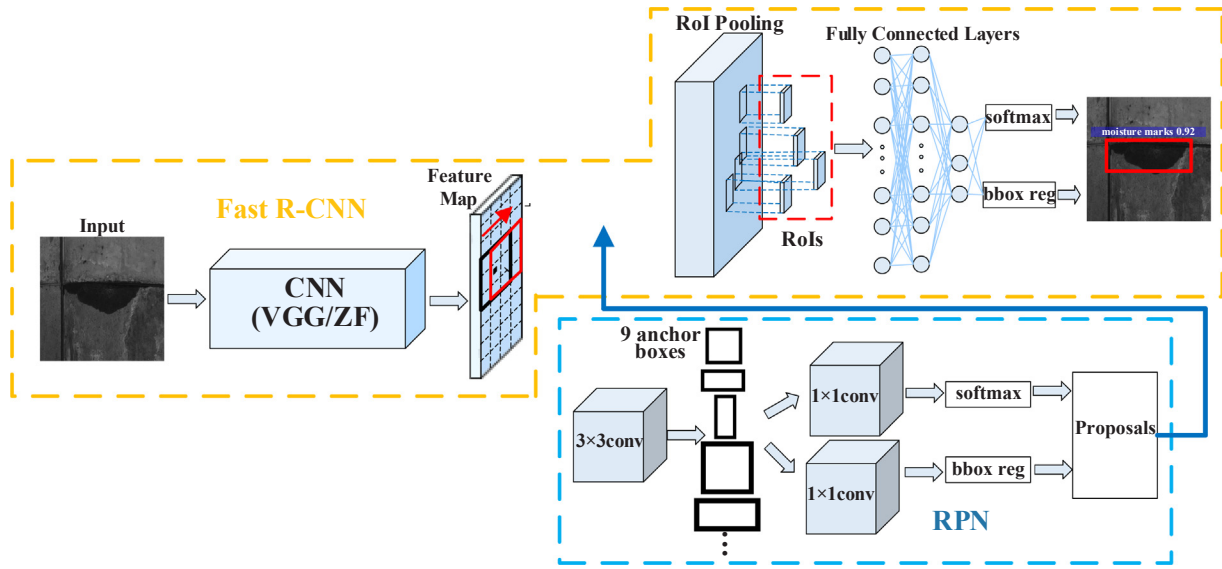For bounding-box regression, the loss is expressed as

Fig. 8. RPN of the Faster R-CNN with a traditional CNN backbone architecture.

$$L_{box}(t^u, v) = \sum_{i \in \{x,y,w,h\}} smooth_{L_1}(t_i^u - v_i) \qquad (2)$$

where

$$smooth_{L_1}(t_i^u - v_i) = \begin{cases} 0.5(t_i^u - v_i)^2 & if \ |t_i^u - v_i| < 1 \\ |t_i^u - v_i| - 0.5 & otherwise \end{cases} \qquad (3)$$

where t_i^u is a vector representing the four parameterised coordinates of the predicted bounding box for class u and v_i is that for the ground-truth box. A detailed description of L_{box} and L_{cls} can be found in a paper by Girshick (2015).

A per-pixel sigmoid is applied to predict the class label, which is used to select the output mask, and L_{mask} is defined as the average binary cross-entropy loss. This definition of L_{mask} allows the network to generate masks for every class without competition among classes and is key for good instance segmentation results (He et al., 2017).

3.4. Training the Mask R-CNN model

In order to modify and train a Mask R-CNN model for the image instance segmentation of moisture marks, experiments were conducted on a computer equipped with one Intel Core i7-5820K central processing unit (CPU), 64 GB of random access memory (RAM), and two GeForce GTX 1080 GPUs (24 GB of graphics memory). The proposed method was implemented based on Detectron, which is a software system by Facebook AI Research that implements state-of-the-art object detection algorithms. The calculation software environment was set with Python 2.7.14, CUDA 8.0, and cuDNN6.0.

Implementation details of the proposed method. An input image was re-sized such that its shorter side has 800 pixels. Synchronous stochastic gradient descent (SGD) (LeCun et al., 1998) was used to train the model on two GPUs. Each mini-batch involves two images per GPU and 256 RoIs per image. The weight decay and momentum were set to 0.0001 and 0.9, respectively. The learning rate was 0.005 for the first 30,000 iterations and 0.0005 for the next 10,000 iterations, and was decreased by 10 at 40,000 iterations. The numbers of RoIs per image used for training and testing were 256 and 1000, respectively. The initial loss was 6.909. This value sharply decreased to 1.968 after 20 iterations and then slowly decreased after 20,000 iterations. After about 45,000 iterations per GPU, the loss function converged to 0.024, as shown in Fig. 11. As a consequence, a set of optimal weights was obtained and saved, i.e. the trained model was obtained. All the
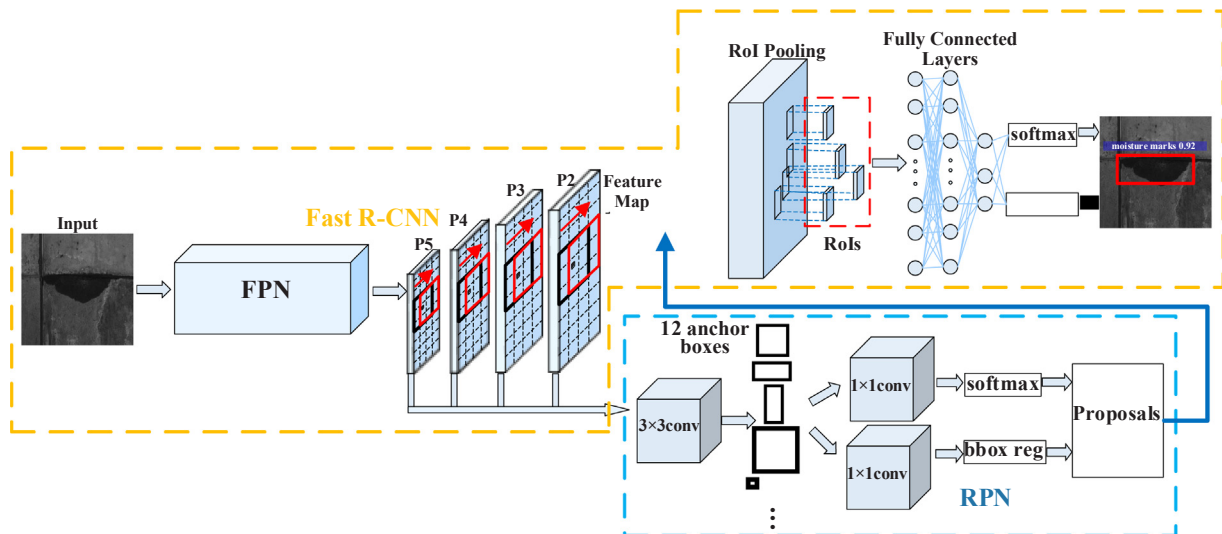

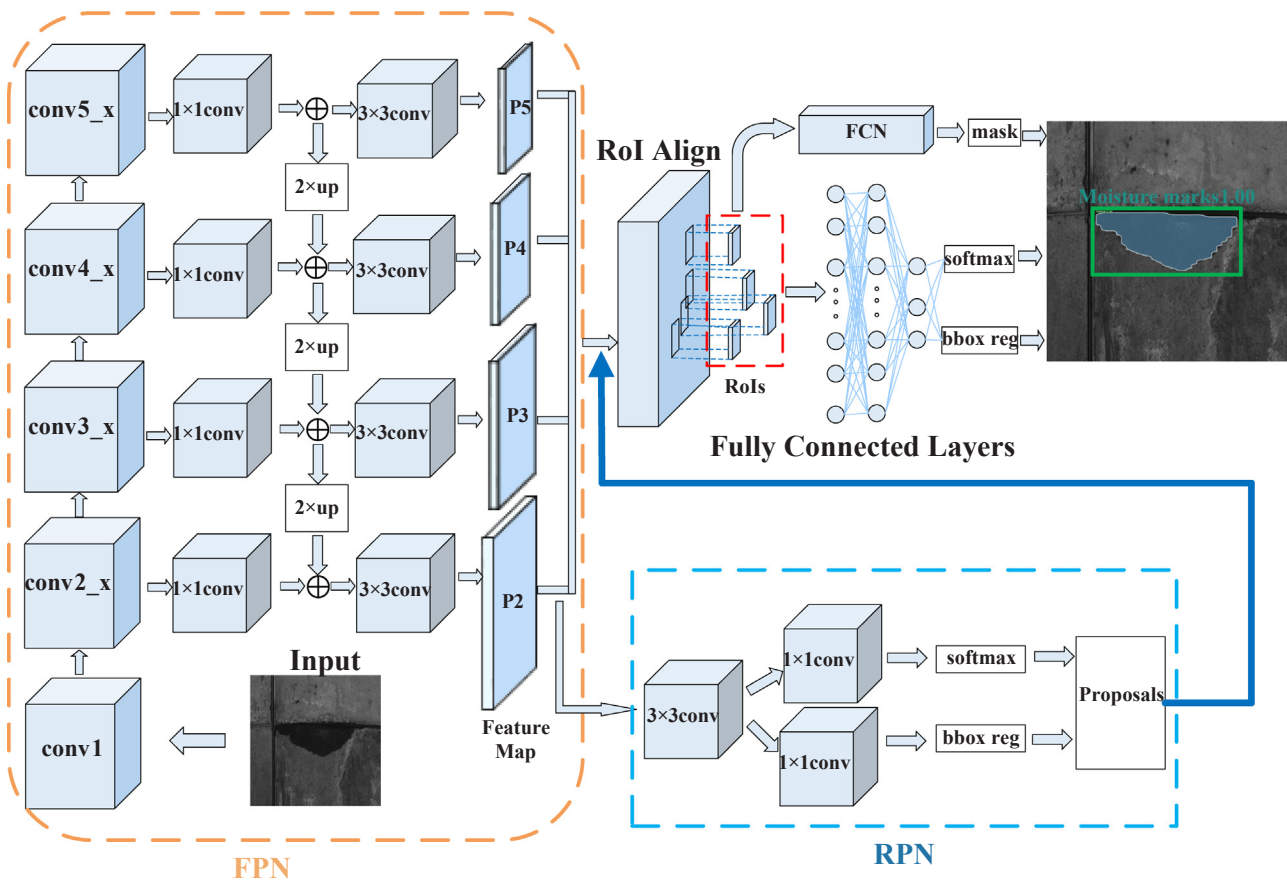**Fig. 9.** RPN of the Faster R-CNN with an FPN backbone architecture.

Fig. 10. Overall architecture of the Mask R-CNN.

aforementioned parameters were using trial and error.

The source code for the Mask R-CNN was modified in order to compute the area of the moisture marks. First, the FCN branch of the trained Mask R-CNN output a closed polygon for the moisture marks in the input image. Second, the values of pixels inside the polygon were assigned values of 1, and the rest were assigned values of 0 to generate a binary image before generating an image with the moisture marks overlaid with a mask. Third, the moisture-mark area was calculated by counting the total number of pixels with a value of 1 in the binary

image, as illustrated in Fig. 12.

4. Experimental evaluation

To examine the performance of the trained and modified Mask R-CNN, 503 new images that were not used for training and validation were used. The performance of the proposed method is also compared with an FCN and two frequently used traditional methods: the RGA and OA. An FCN is a framework that can perform semantic segmentation
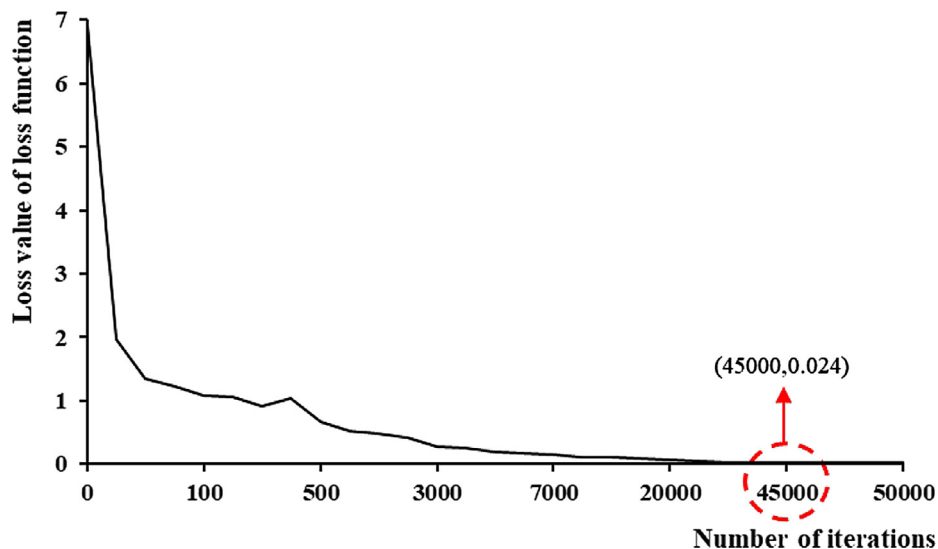

Fig. 11. Learning curve of the training process.

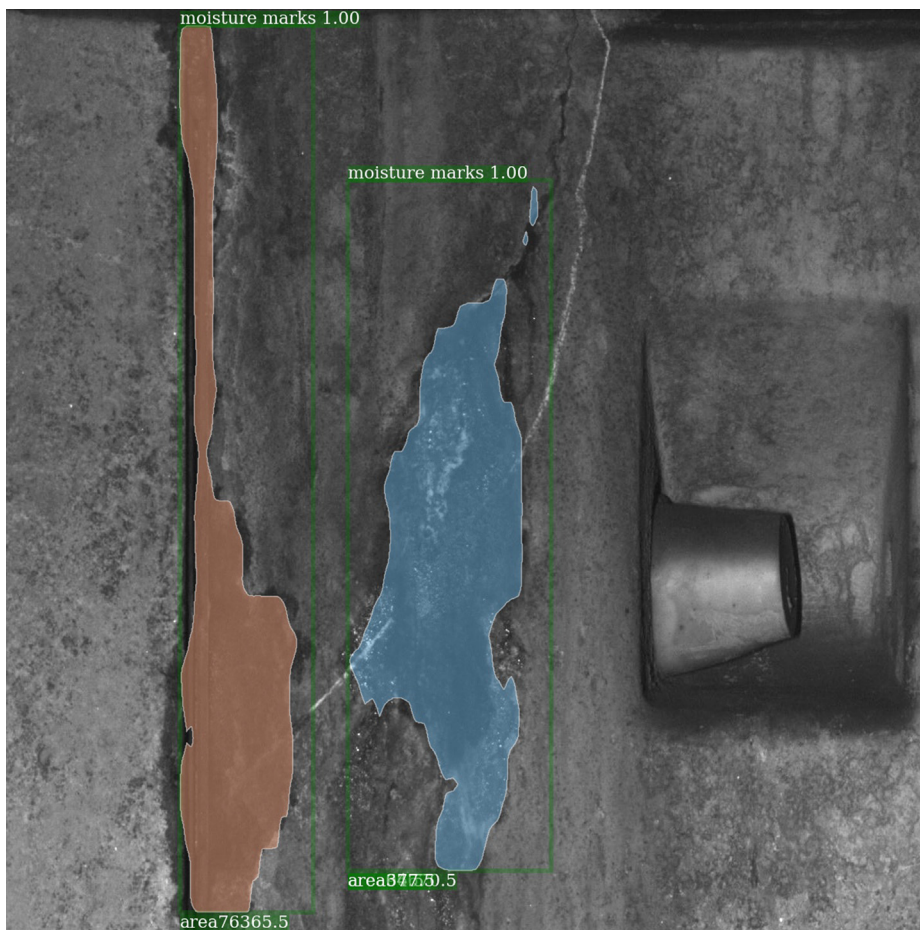**Fig. 12.** Quantification of moisture marks.

Fig. 13. Results of the segmentation of moisture marks by the Mask R-CNN.

tasks with efficient learning and inference. A detailed description of its advantages can be found in a paper by Long et al. (2015). The RGA is a classical region-based image segmentation method that examines the neighbouring pixels of initial seed points and determines whether these neighbouring pixels should be added to the region (Kamdi and Krishna, 2012; Huang et al., 2018). The OA is a method that segments an image into a foreground (e.g. moisture marks) and background via a threshold. When the best segmentation threshold is used, the background should differ the most from the foreground. The OA finds the best segmentation threshold by the maximum between-cluster variance (the maximum variance between the pixels' grey values) (Otsu, 1979).

4.1. Recognition results

The new images were fed into the trained Mask R-CNN. Before overlaying a mask on the moisture marks in an image, the trained model output a binary image. A value of 0 defines a non-moisture-mark region, whereas a value of 1 represents the moisture-mark region. After counting the total number of pixels with a value of 1, the moisture-mark areas were printed at the bottom-left of the bounding box, as illustrated in Fig. 13.

Fig. 14 shows part of the recognition results of the test images. The results in Fig. 14 shows that the proposed Mask R-CNN provides clear moisture-mark information. Although the RGA and OA provide some moisture-mark information, they cannot provide more information because of the effects of circumferential joints, longitudinal joints, bolt holes, etc. Comparing the authors' previous FCN model (Huang et al., 2018) and the proposed Mask R-CNN model, both the Mask R-CNN and FCN models provide clear moisture-mars information. However, the concrete stain in the raw image (Fig. 14) is incorrectly identified as a moisture mark by the previous FCN model.

In general, the experiment shows that the performance of the RGA and OA is quite dependent on the image conditions. In contrast, the proposed Mask R-CNN works quite well regardless of the image
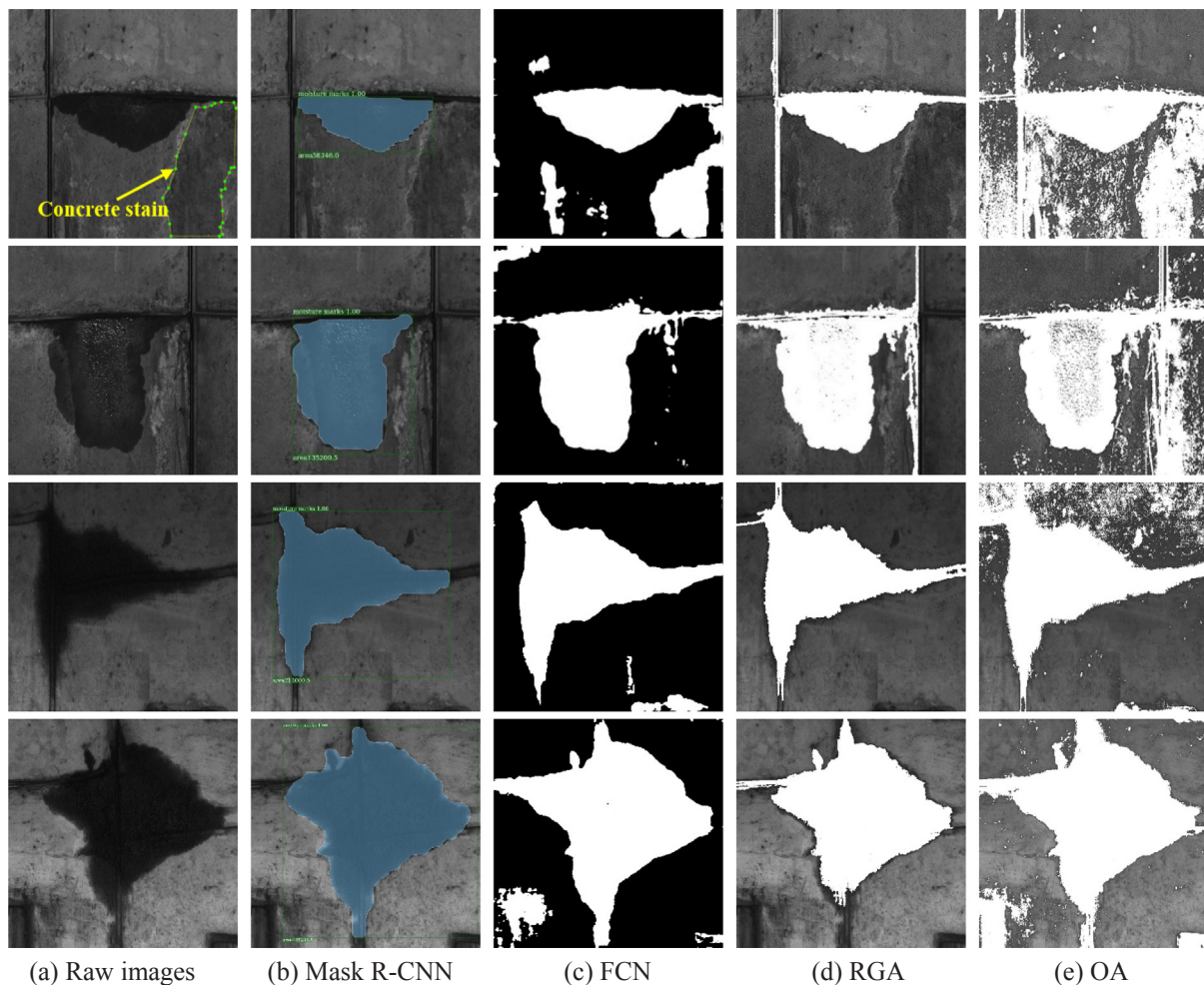
(a) Raw images   (b) Mask R-CNN   (c) FCN   (d) RGA   (e) OA

Fig. 14. Results of the segmentation of moisture marks by different algorithms.

Table 1
Metrics of different algorithms for segmenting moisture marks.

Method	Accuracy	F_1 score	IoU
RGA	0.9190	0.7757	0.6621
OA	0.7143	0.5027	0.3579
Previous FCN	0.9481	0.8952	0.8103
Proposed	0.9846	0.9466	0.9004

Table 2
Inference times of different algorithms for segmenting moisture marks.

Method	Inference time (s/image)
RGA	30.637
OA	0.081
Previous FCN	1.10
Proposed	0.107

conditions.

4.2. Metrics of the algorithms for segmentation

In this study, three metrics, i.e. the accuracy, F_1 score (Fawcett, 2006), and IoU (called IU in Long et al., 2015), are used to evaluate the performance of different models. The event where a moisture-mark pixel is segmented as a moisture-mark pixel is denoted as a true positive (TP), the event where the background pixel is segmented as the background pixel is denoted as a true negative (TN), the event where the background pixel is segmented as a moisture-mark pixel is denoted as a false positive (FP), and the event where a moisture-mark pixel is segmented as the background pixel is denoted as a false negative (FN). Thus, the three metrics are computed as follows:

accuracy: $(TP + TN)/(TP + TN + FP + FN)$
precision: $TP/(TP + FP)$
recall: $TP/(TP + FN)$
F_1 score: $2/(1/precision + 1/recall)$
IoU: $TP/(TP + FP + FN)$

where *TP*, *TN*, *FP*, and *FN* are the numbers of TPs, TNs, FPs, and FNs, respectively.

The results of a comparison for the 503 test images are presented in Table 1. The performance of the proposed method is quite significant with an accuracy of 98.46%, which is much higher than those for the RGA (91.90%) and OA (71.43%). The proposed method attains an F_1 score of 94.66%, which is much higher than those for the RGA (77.57%) and OA (50.27%). In terms of the IoU, the differences should be particularly emphasised: the proposed method has an IoU of 90.04% compared to 66.21% for the RGA and 35.79% for the OA. Comparing the previous FCN algorithm with the proposed Mask R-CNN algorithm, the accuracy, F_1 score, and IoU of the previous FCN (94.81%, 89.52%, and 81.03%) are slightly lower than those for the proposed Mask R-CNN (98.46%, 94.66%, and 90.04%). In summary, the proposed method has better performance for moisture-mark instance
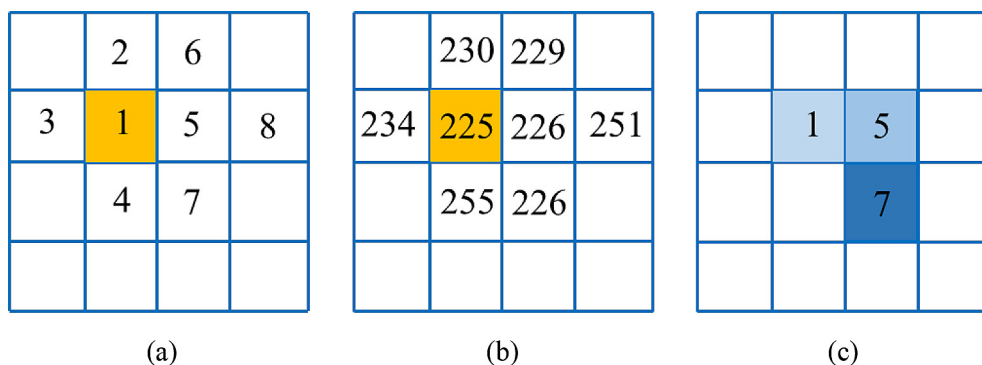
Fig. 15. Region growing process: (a) marking a number of pixels, (b) corresponding grey values of the marked pixels, and (c) the direction of region growth.

segmentation.

4.3. Inference time

If there is a large number of lining images, the total inference time will be a crucial factor to consider. The inference times of four algorithms are presented in Table 2. The inference time of the RGA is affected by the image size; a larger image size results in a longer computation time. Moreover, the inference time of the RGA is much longer than those of the OA and proposed method.

The OA method has a low inference time but sacrifices the accuracy, F_1 score, and IoU. The inference time of the proposed method is only 1.3 times higher than that of the OA, but the accuracy, F_1 score, and IoU are about 1.4, 1.9 and 2.5 times as high as those of the OA, respectively.

The inference time of the proposed Mask R-CNN includes the times for detecting the bounding box, detecting the mask, and adding the mask to an image. Despite this, the inference time of the previous FCN (1.10 s per image) is a little over 10 times longer than that of the proposed Mask R-CNN. Thus, the proposed Mask R-CNN method is more accurate and rapid than the FCN method for segmenting the moisture marks in images.

The proposed method and previous FCN compute via GPUs, which are very good for image data processing owing to the parallel computation of numerous vectors and matrices. However, the RGA and OA usually compute via CPUs (Cha et al., 2017a; Yu et al., 2017). Thus, when the background noise of an image is more complicated, its corresponding computation time is relatively long.

5. Discussion

The differences in the recognition results among the four algorithms are discussed in this section. The RGA is a method that examines the neighbouring pixels of the initial seed points and determines whether these neighbouring pixel should be added to the region. The process is shown in Fig. 15. Point 1 is first selected as the seed point (Fig. 15(a)). The threshold difference between this seed point and the neighbouring pixels is 1. Then, Point 5 is added to the segmentation region and selected as next seed point because the grey value of Point 5 is closest to that of the seed point among the neighbouring points (Points 2–5) (Fig. 15(b)). If the threshold difference is less than 1, the region does not grow. Now, the mean grey value of the segmentation region is 225.5. Therefore, Point 7 will be selected as the next seed point during the second cycle (Fig. 15(c)). This process is carried out until the difference between the next neighbouring point and the segmentation region is more than 1. The RGA largely depends on the selection of the first seed point and the grey values of the neighbouring pixels. However, the grey values of moisture marks are nearly the same as those of circumferential joints, longitudinal joints, and bolt holes. As a result, segmental joints and bolt holes may be recognised as moisture marks.

Segmentation with the OA is also based on the grey-value characteristics. It finds the best segmentation threshold by maximising the variance between pixels' grey values. Because it is influenced by the segmental joints and other objects in an image, the OA has a poor performance for identifying moisture marks. This indicates that separating the moisture marks from the background only through the grey value of a pixel results in poor performance.

An FCN is trained end-to-end and pixels-to-pixel on semantic segmentation by efficient dense feedforward computation and backpropagation. It can be combined with a Faster R-CNN, forming a unified Mask R-CNN framework to perform instance segmentation masks.

Unlike the RGA and OA, the proposed Mask R-CNN computes the features of moisture marks through self-learning, which is an iterative process in which the loss function is decreased by the SGD algorithm. This process is iterated until the loss function converges and a set of optimal weights is obtained and saved. In the test process, the obtained weights are directly used to predict moisture marks. Therefore, it is not surprising that the proposed method is able to achieve good results.

6. Conclusions

This paper provides a method for the image instance segmentation of moisture marks of shield tunnel lining. Instance segmentation is carried out in two main steps. First, the segmentation dataset must be created. The images used to create the dataset utilised in this study were acquired by MTI-200a. The images were annotated by LabelMe, and the annotation file was converted into a format similar to that of the COCO dataset through Python. Second, the Mask R-CNN needs to be modified and trained to identify moisture marks. The hyperparameters suitable for specific problems can be obtained from experiments, and the source code for the Mask R-CNN was modified in order to compute the area of the moisture marks. The modified Mask R-CNN was then trained on the created dataset until the loss function converged.

For 503 test images, the proposed method achieves a relatively low inference time but has much better performance with regards to the accuracy, F_1 score, and IoU, which are all higher than those of two frequently used methods—the RGA and OA. The accuracy, precision, and recall of the proposed Mask R-CNN method are also slightly higher than those of the FCN method with a lower inference time. The modified algorithm based on the Mask R-CNN is helpful for rapidly and accurately recognising the moisture marks of shield tunnel lining in terms of the accuracy, precision, recall, and inference time.

However, the above results are based on a training and validation dataset with a total of 4,528 images. In the future, more images can be acquired by MTI-200a. The TLWLCO database can be enlarged to further increase the accuracy and robustness of the proposed method. Further, image instance segmentation for cracks will be realised by combining the responses at the Mask R-CNN's FCN branch with a fully connected conditional random field (CRF).

Acknowledgements

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.tust.2019.103156.

References

Abdel-Qader, I., Abudayyeh, O., Kelly, M.E., 2003. Analysis of edge-detection techniques for crack identification in bridges. J. Comput. Civil. Eng. 17 (4), 255–263.

Ai, Q., Yuan, Y., Bi, X., 2016. Acquiring sectional profile of metro tunnels using charge-coupled device cameras. Struct. Infrastruct. Eng. 12 (9), 1065–1075.

Cha, Y.J., Choi, W., Büyüköztürk, O., 2017a. Deep learning-based crack damage detection using convolutional neural networks. Comput.-Aided Civ. Infrastruct. Eng. 32 (5), 361–378.

Cha, Y.J., Choi, W., Suh, G., Mahmoudkhani, S., Büyüköztürk, O., 2017b. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Comput.-Aided Civ. Infrastruct. Eng. 33 (9), 731–747.

Dai, J., Li, Y., He, K., Sun, J., 2016. R-FCN: object detection via region-based fully convolutional networks. arXiv: 1605. 06409, pp. 1–11.

Dawood, T., Zhu, Z.H., Zayed, T., 2018. Computer vision-based model for moisture marks detection and recognition in subway networks. J. Comput. Civ. Eng. 2018, 32(2): 04017079.

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recogn. Lett. 27, 861–864.

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. Adv. Eng. Inform. 26 (4), 846–858.

Girshick, R., 2015. Fast R-CNN. arXiv: 1504.08083.

He, K.M., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask R-CNN. arXiv: 1703. 06870, pp. 1–10.

Hu, C.P., Zhu, H.H., Li, X.J., 2010. Detection of tunnel water leakage based on image processing. Inform. Technol. Geo-Eng. 254–262.

Huang, H.W., Shao, H., Zhang, D.M., Wang, F., 2017a. Deformational responses of operated shield tunnel to extreme surcharge: a case study. Struct. Infrastruct. Eng. 13 (3), 345–360.

Huang, H.W., Zhang, Y.J., Zhang, D.M., Ayyub, B.M., 2017b. Field data-based probabilistic assessment on degradation of deformational performance for shield tunnel in soft clay. Tunn. Undergr. Space Technol. 67, 107–119.

Huang, H.W., Sun, Y., Xue, Y.D., Wang, F., 2017c. Inspection equipment study for subway tunnel defects by grey-scale image processing. Adv. Eng. Inform. 32, 188–201.

Huang, H.W., Li, Q.T., Zhang, D.M., 2018. Deep learning based image recognition for crack and leakage defects of metro shield tunnel. Tunn. Undergr. Space Technol. 77, 166–176.

Kamdi, S., Krishna, R.K., 2012. Image segmentation and region growing algorithm. Int. J. Comput. Technol. Electron. Eng. 1 (2), 103–107.

LeCun, Y., Bottou, L., Orr, G.B., Müller, K.-R., 1998. Efficient backprop. Neural Networks: Tricks of the Trade, LNCS 1524, 9–50.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L., 2015. Microsoft COCO: Common Objects in Context. arXiv: 1405.0312, p. 1–12.

Lin, T.-Y., Dollár, P., Girshick, R., He, K.M., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection. arXiv: 1612.03144, pp. 1–10.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: The 28th IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA, pp. 3431–3440.

Lu, S.Z., Ren, C.L., Zhang, J.X., Zhai, Q., Liu, W., 2018. A novel approach to droplet's 3d shape recovery based on Mask R-CNN and improved lambert–phong model. Micromachines 9, 1–12.

Nishikawa, T., Yoshida, J., Sugiyama, T., Fujino, Y., 2012. Concrete crack detection by multiple sequential image filtering. Comput.-Aided Civ. Infrastruct. Eng. 27 (1), 29–47.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybernet. SMC-9 (1), 62–66.

Ren, S.Q., He, K.M., Girshick, R., Sun, J., 2017. Faster RCNN: towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. 39 (6), 1137–1149.

Shahin, M.A., Maier, H.R., Jaksa, M.B., 2004. Data division for developing neural networks applied to geotechnical engineering. J. Comput. Civ. Eng. vol.18, 2, pp. 105–114.

Soukup, D., Huber-Mörk, R., 2014. Convolutional neural networks for steel surface defect detection from photometric stereo images. In: The 10th International Symposium on Visual Computing, Las Vegas, NV, pp. 668–677.

Tan, K., Cheng, X., Ju, Q., Wu, S., 2016. Correction of mobile tls intensity data for water leakage spots detection in metro tunnels. IEEE Geosci. Remote Sens. Lett. 13, 1711–1715.

Ukai, M., Nagamine, N., 2011. A high-performance inspection system of tunnel wall deformation using continuous scan image. The 9th World Congress on Railway Research, Lille, France. < http://www.railway-research.org/IMG/pdf/poster_ukai_masato.pdf > .

Wada, K. Labelme: image polygonal annotation with python. Retrieved from. https://github.com/wkentaro/labelme.

Wang, R.L., Xiao, T.G., Zhu, Y., 2011. Water leakage treatment and deformation control of shield tunnel in Shanghai metro. Undergr. Eng. Tunn. Supp. 2, 102–108 in Chinese.

Wu, C.R., Huang, H.W., 2018. Laser scanning inspection method and application for metro tunnel leakage. J. Nat. Disast. 27 (4), 59–66 in Chinese.

Xu, T., Xu, L., Li, X., Yao, J., 2018. Detection of water leakage in underground tunnels using corrected intensity data and 3d point cloud of terrestrial laser scanning. IEEE Access 6, 32471–32480.

Xue, Y.D., Li, Y.C., 2018. A fast detection method via region-based fully convolutional neural networks for shield tunnel lining defects. Comput.-Aided Civ. Infrastruct. Eng. 33 (8), 638–654.

Yu, T., Zhu, A., Chen, Y., 2017. Efficient crack detection method for tunnel lining surface cracks based on infrared images. J. Comput. Civil Eng. 31 (3), 1–11.

Yu, P., Wu, H., Liu, C., Xu, Z., 2018. Water leakage diagnosis in metro tunnels by intergration of laser point cloud and infrared thermal imaging. ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, pp. 2167–2171.

Zhang, W.X., Witharana, C., Liljedahl, A.K., Kanevskiy, M., 2018. Deep convolutional neural networks for automated characterization of arctic ice-wedge polygons in very high spatial resolution aerial imagery. Remote Sens. 10 (9), 1–31.

Zurowietz, M., Langenkämper, D., Hosking, B., Ruhl, H.A., Nattkemper, T.W., 2018. MAIA—a machine learning assisted image annotation method for environmental monitoring and exploration. PLoS One 13 (11), 1–18.