SMAR 2019 – Fifth Conference on Smart Monitoring,
Assessment and Rehabilitation of Civil Structures

SMAR 2019

Application of deep learning-based crack assessment technique to civil structures

Soojin CHO [1], Byunghyun KIM [1], Geonsoon KIM [1]
[1] University of Seoul, Seoul, South Korea

Contact e-mail: soojin@uos.ac.kr

ABSTRACT: In this study, a deep learning-based automated crack detection technique has been applied to real structures. The deep learning model used in this study is a Mask R-CNN model pre-trained with 200,000 COCO dataset, and a transfer learning is performed using training data collected from Internet and the other bridges. Various types of cracks are marked as ground truths on more than 1,000 images with 1000×1000 pixel resolution for the training, and they are used for the transfer learning. The trained model is developed to distinguish cracks from the concrete surface, especially against similar objects that exhibit color and contrast features similar to those of cracks. The developed model is applied to three types of civil structures, e.g., a bridge, a tunnel, and a concrete road, and the detection results are analyzed in depth using performance measures. The result shows that the carefully trained deep learning model can work as an effective alternative to the current visual inspection.

1 GENERAL INSTRUCTIONS

Given that concrete is the major construction material worldwide, cracks need to be assessed during the maintenance of concrete structures for mainly two reasons: durability and aesthetics. Crack is a crucial index visually showing loading condition of a structure. Furthermore, crack can expose rebars in concrete to water and air, resulting in the corrosion and further deterioration. Wide cracks may attract the attention of people and make them uncomfortable, regardless of mechanical effects to structural safety. Despite of the importance, however, current crack assessment is mostly dependent on visual inspection that has some drawbacks, such as subjective decision according to inspector's expertise, high labor intensity, expensive logistic time, etc.

To overcome the drawback with the support of ICT techniques, civil engineers put a lot of efforts to replace visual inspection with computer vision (CV) inspection. The CV inspection is actively being implemented to steel corrosion detection (Valeti and Pakzad, 2017), concrete spalling detection (German, Brilakis and DesRoches, 2012). Furthermore, many researchers attempted to develop CV inspection methods to assess cracks that has the smallest visibility among the concrete damages (Hutchinson and Chen, 2010). But the CV inspection methods for cracks are still in the laboratory, since most of them are developed under idealized environment.

Rapid development of deep learning based on convolutional neural networks (CNNs) is now solving many difficulties in the real-world CV techniques. The CNN models such as ResNet (He *et al.*, 2016) enabled automated feature extraction to classify various kinds of objects in images. Answering the rapid development of deep learning, many attempts have been made to implement deep learning to detect cracks. Yang *et al.* (2018) detected crack using a FCN and measured the width of cracks on the pixel level. Kim and Cho (2018a) detected cracks using 5-

classed CNN on concrete surfaces of the real-world environment. Though the literatures have shown the deep learning leveraged the detectability of cracks, their applicability to the structural inspection has not been fully validated due to the complexity of real structure environment.

Beyond the limitations, a deep learning-based automated crack detection technique is developed using a Mask and Region-based Convolutional Neural Network (Mask R-CNN) (He *et al.*, 2017), which has been pre-trained with common objects in context (COCO) dataset (Lin *et al.*, 2014). Then the model is fine-tuned using a transfer learning approach using crack images collected from Internet and the other structures. The trained model is able to distinguish cracks from the concrete surface, especially against similar objects that exhibit color and contrast features similar to those of cracks. The technique is applied to three types of civil structures, e.g., a bridge, a tunnel, and a concrete road, and the detection results are analyzed in depth using performance measures. For a bridge, the detected cracks parts are processed to assess the crack information such as crack width and length.

2 MASK AND REGION-BASED CONVOLUTIONAL NEURAL NETWORK (MASK R-CNN)

2.1 Architecture

Convolutional neural network (CNN) is one of the many other types of neural networks and constrain the architecture of image input in the form of 3D volumes of neurons. Most of CNN consists of convolutional layer, max pooling layer, fully connected layer and softmax layer. But combination of the components of CNN can vary according to the purpose of detection and the form of input and output. For instance, Faster Region-based CNN (Faster R-CNN) (Ren *et al.*, 2017) is developed to localize objects in an image using rectangular bounding boxes. Fully-convolutional network (FCN) (Long, Shelhamer and Darrell, 2015) that consists of only convolutional layers is developed to conduct pixel-to-pixel object masking. Mask R-CNN can be referred to combination of Faster RCNN and FCN and conduct object localization and object masking simultaneously as shown in Figure 1.

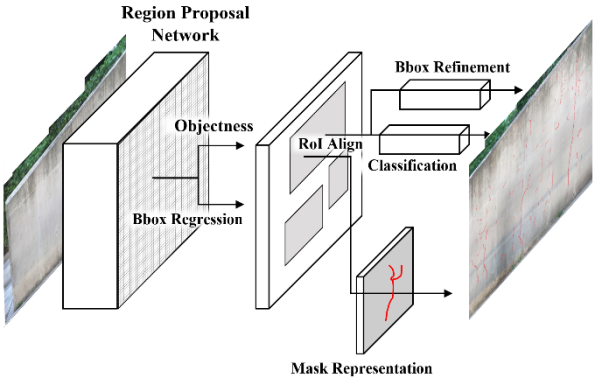


Figure 1. Overall arcitecture of Mask R-CNN

In the first step of Mask R-CNN, region-proposal network (RPN) finds the possible objects in an image. RPN is a FCN model which outputs an objectness score and a group of object proposals with bounding-boxes for an input image of any size. To generate region proposals, a small window slides over the convolutional feature map extracted by the last convolutional

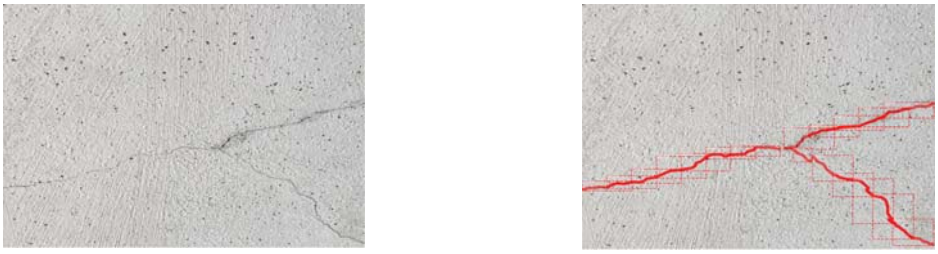
layer. Each array obtained by sliding window is converted to a lower-dimensional feature and the feature is fed into two fully convolutional layer—a box-regression layer and a box-classification layer. Each sliding-window location has k possible proposals at maximum. The box-regression layer which depicts the coordinates of the box with x, y, width and height has 4k outputs and the box-classification layer which is a two-class softmax layer has 2k outputs as scores estimating probability of object or not object for each proposal.

In the second step, Mask R-CNN takes the object proposal results from RPN and extracts features of the objects using RoIAlign layer from each candidate box and performs classification. RoIAlign creates a w × h size feature map (7 × 7 in this paper) from the RoI extracted by RPN. In this process, being expressed in real numbers, the location of RoI cannot be precisely expressed on the convolutional layer which is expressed in integer. Simply rounding the location of RoI to integer may result in poor feature extraction result. To avoid the loss of precise location of ROI, RoIAlign layer computes the exact value of each sampling point by bilinear interpolation from the nearby points on the feature map. The feature map extracted by RoIAlign layer is fed to classification layer and bounding-box refinement layer. The feature maps are flattened in the form of a fully connected layer before being fed into the classification and bounding-box regressor layers. The class of the object is determined in the classification layer. The bounding-box regressor layer work in the very similar way as RPN works and it refines the location and size of the bounding box.

In the third step, the feature map extracted by RoIAlign layer in the second step is also fed into mask branch to decide the spatial layout of the object in the bounding box. The mask branch is a FCN which segments the object in an image pixel by pixel. The output size of the mask branch is 56 × 56 and the branch network follows the FPN mask branch network of the paper. The mask branch parallel to the classification layer is able to predict the mask without the object class specified by the classification layer.

2.2 Training

To train a Mask R-CNN model, 319 crack images were collected from the Internet or captured from real structures. The resolution of each image is slightly different, but the resolutions of all images are ranged from 600 × 600 to 2000 × 2000. Photoshop was used as an annotation tool to annotate cracks with red as shown in the Figure 2. After annotation of all the training images, the annotation format is converted to that of COCO dataset using image processing. In the figure, the colored parts are the masks of the concrete damage that the mask branch trains, and the dotted lines are bounding boxes for which RPN, box-classification layer, and box-regression layer will be trained. It is worthy to note that the cracks are annotated with overlapped bounding boxes when longer than a certain length. These overlapped bounding boxes prevent excessively large concrete surfaces around the cracks from being included in the bounding boxes.



(a) (b)

Figure 2. Training data examples: (a) original images and (b) annotation display

The experiments are performed using the open source Mask R-CNN which is pre-trained by a COCO dataset (Lin *et al.*, 2014). Since the masking regions of the COCO dataset are set for general objects with appropriate areas, the data input is modified to consider slender regions, as shown in Figure 1, for cracks. The training was carried out on a workstation with 2 Intel Xeon processors and 4 NVIDIA GTX 1080 Ti. Each mini-batch has 1 image per GPU and the training images are resized to 800 pixels in shorter edge if the size is smaller or longer than that following the parameters of the work of He *et al.* (2017). The network is trained for a total of 50 epochs at 96 iterations per epoch. Learning rate, weight decay, and momentum are set as 0.001, 0.0001 and 0.9 respectively.

3 APPLICATION TO CIVIL STRUCTURES

3.1 Test Structures

The developed method was applied to three structures: a bridge, a tunnel, and pavement of a concrete road. The test bridge is an 8-span PSC box girder bridge, and the appearance is imaged using a commercial drone (DJI Inspire 2) equipped with a high-resolution camera whose resolution is 6616 × 4008 pixels. Since any damage is not found on the girder, the method was applied on the images from the pier. The test tunnel was scanned using a tunnel imaging car specially designed by the tunnel authority. The imaging car has five cameras with resolution of 4112 × 2176 pixels and multiple infra-red (IR) lights considering low illumination inside the tunnel. The pavement was scanned using a scanning car designed by the road authority. The car has a line-scanner on the back and the line-scanned image was post-processed to build 2D image with resolution of 5000 × 13450 pixels. Figure 3 shows the imaging devices used for the bridge, the tunnel, and the pavement, respectively. All the obtained images on the structure were segmented into 1024 x 1024 pixels to input the mask R-CNN model. For each structure, 10 images with cracks and other interrupting objects are used to validate the performance of the proposed method at in-situ condition.

Note that the ground-truth crack widths could not be identified, since no visual inspections were carried out on the structure. However, based on the inspection reports of similar structures, the range of crack widths could be assumed: 0.3-1.0mm for the bridge piers, 0.5-1.0mm for the tunnel, and 0.1-1.0mm for the pavement. Instead, The detectability of the proposed method can be referred to Kim and Cho (2019) which tested on a concrete wall with hundreds of cracks whose widths range 0.1-1.0mm.

(a) (b) (c)

Figure 3. Imaging equipment used for crack detection: (a) DJI inspire 2 drone, (b) tunnel imaging car and (c) road scanning car

3.2 Example Results

Figure 4-6 shows the example results of implementing the proposed crack detection method to the images taken from the test bridge, tunnel and concrete road respectively. In Figure 4-6, the upper row shows the examples of exact crack detection and lower row shows the examples of relatively inaccurate crack detection result. Also, the left column of Figure 4-6 shows the original images captured by imaging equipment and the right column shows the crack detection result by the trained Mask R-CNN. Throughout Figure 4-6, all the pixels predicted as cracks by the trained Mask R-CNN is highlighted with red color. And each prediction result is compared to the original images to check if the prediction result is correct or not. The comparison result is classified into three categories, namely true positive, false positive and false negative. True positive, which is marked with red color in Figure 4-6, is a detection result where the model correctly predicts the existing cracks. False positive marked with blue colored letters and dashed boxes is a detection result where the model incorrectly predicts the cracks. False negative is marked with greed colored letters and dashed boxes missed cracks by the model.

In Figure 4, the trained Mask R-CNN successfully detected cracks even though the cracks on the bridge pier have been captured into two or three pixels in width. The most confusing objects on the bridge pier is the marks printed by concrete formworks. In the exact example of Figure 4, the formwork mark is not sharp enough to be determined as a crack, but in the example of inaccurate result small parts of the formwork mark is determined as crack due to its similarity to crack. In Figure 5, some of spider web is determined as crack. Old spider webs are left with only a few lines, so they lose their original shape and become looking like a crack. Even though most of spider webs are successfully removed but some of them are determined as crack. In Figure 6, the trained Mask R-CNN successfully detected most of the cracks, but a few cracks were missed due to their narrow width.

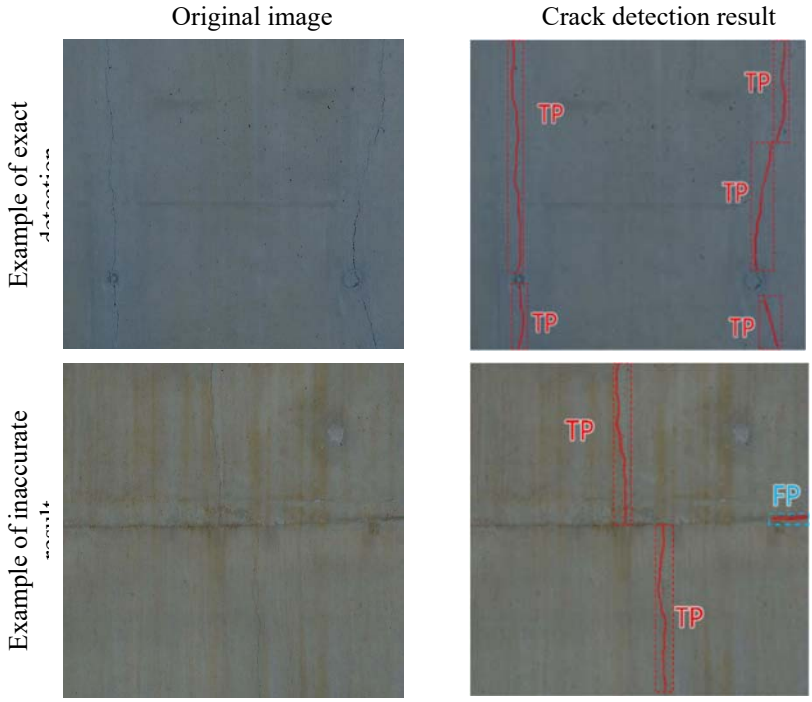


Figure 4. Example results from a test bridge
(TP: True Positive, FP: False Positive, and FN: False Negative)

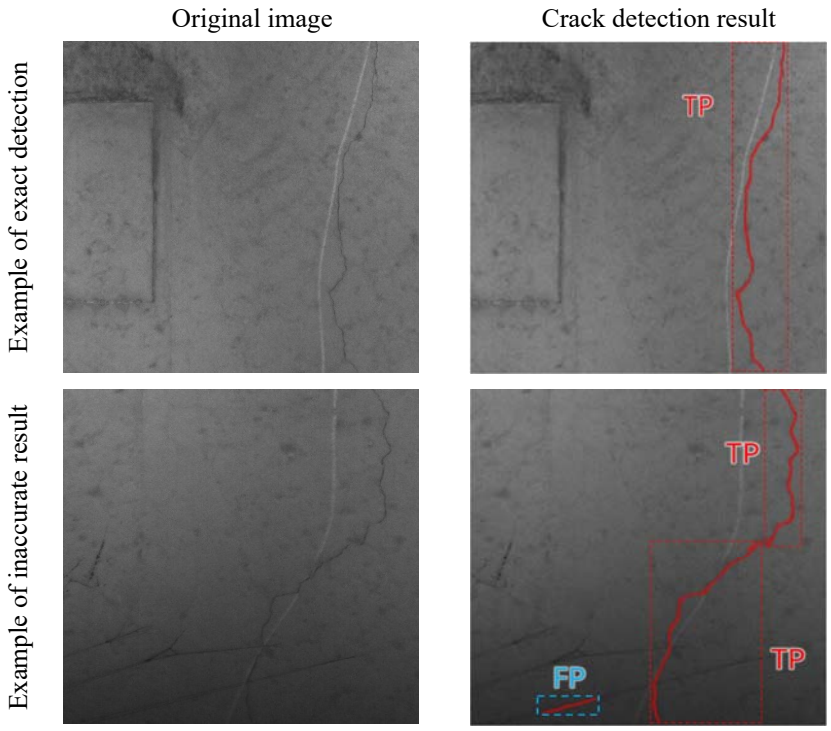


Figure 5. Example results from a test tunnel
(TP: True Positive, FP: False Positive, and FN: False Negative)

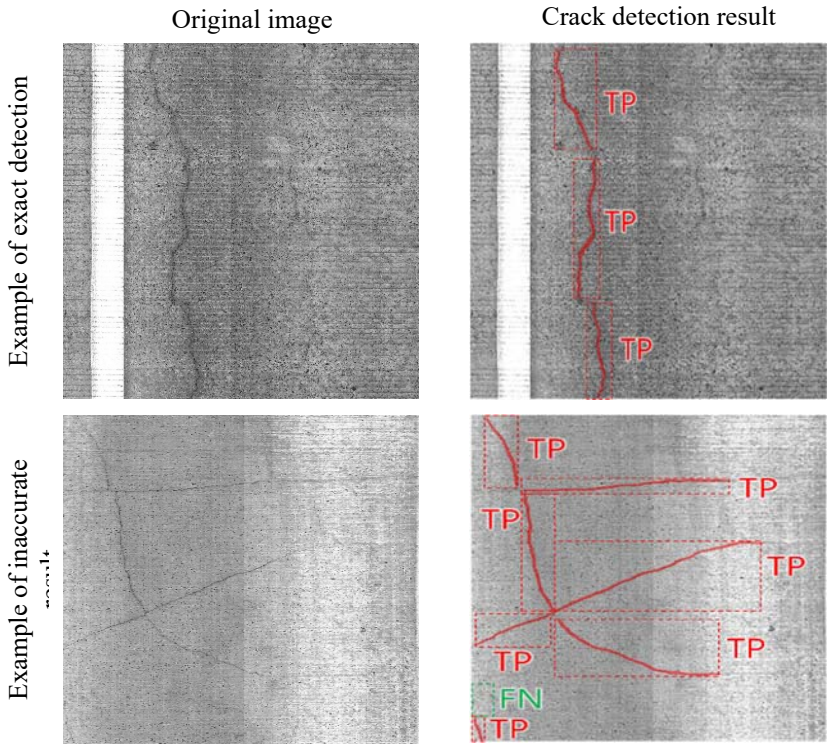


Figure 6. Example results from a test tunnel
(TP: True Positive, FP: False Positive, and FN: False Negative)

3.3 Performance Measure

Table 1 shows the performance measure (i.e., precision and recall) of the proposed method for 10 images from each structure (30 images total). In Table 1, there are four categories classifying crack detection results, namely existing cracks, detected cracks, missing cracks and false detection which refer to ground truth, true positives, false negatives and false positives respectively. Precision is the number of detected cracks divided by the sum of detected cracks and false detections given as a percentage. Recall is the number of existing cracks divided by the sum of detected cracks and missing cracks given as a percentage. Precision shows how accurate the detection of the trained Mask R-CNN is and Recall shows how many existing cracks are found by the trained Mask R-CNN. For the bridge, among 16 existing cracks, 16 were detected and there are 4 false detection resulting 80.0% in precision and 100.0% in recall. For the tunnel, among 13 existing cracks, 13 were detected and there are 4 false detection resulting 76.4% in precision and 100% in recall. For the tunnel, among 45 existing cracks, 42 cracks were detected, and 4 false detections were found resulting in 97.6% precision and 93.3% recall. In total, the precision was estimated as 88.7%, while the recall was as 95.9%, which shows the high applicability of the trained Mask R-CNN to the real CV inspection for concrete structures.

Table 1. Performance measure of proposed method for 10 images from each structure.

	Bridge	Tunnel	Concrete Road	Total
No. of images	10	10	10	30
Existing Cracks	16	13	45	74
Detected Cracks	16	13	42	71
Missing Cracks	0	0	3	3
False Detection	4	4	1	9
Precision (%)	80.0	76.4	97.6	88.7
Recall (%)	100.0	100.0	93.3	95.9

4 CONCLUSION

This study presented a novel concrete crack detection method based on state-of-the-art deep learning model, Mask R-CNN. To train Mask R-CNN for crack detection, 319 crack images collected from the Internet and captured from real structures are used as training data. The annotated images of cracks were converted to the same format as the annotation file of COCO dataset through the additional image processing, and the Mask R-CNN was trained for 50 epochs with a workstation equipped with four GPUs.

The validation of the trained Mask R-CNN model is conducted on a total of 30 images captured from three types of real structures (a bridge pier, a tunnel, and a concrete pavement) using a drone, tunnel imaging car and road scanning car, respectively. The trained Mask R-CNN detected 71 cracks out of 74 existing cracks and missed 3 cracks and brought 9 false detection resulting in average 88.7% in precision and 95.9% in recall. The experimental results showed that the trained Mask R-CNN can detect cracks during real bridge inspection.

The automated concrete crack detection with Mask R-CNN can be considered as a promising technology to replace the existing visual inspection. The trained Mask R-CNN in this study will

be able to perform concrete crack detection with high performance in the real structure images collected using digital camera or unmanned aerial vehicles. In further study, additional dataset collection may increase accuracy of the trained Mask R-CNN in more diverse fields, and the structure of Mask R-CNN can be more lightened to be used in higher resolution images or in small devices such as mobile phones. Furthermore, the types of the damages to be detected can be diversified to spalling, efflorescence, rebar exposure, and segregation after collecting enough image dataset.

ACKNOWLEDGEMENT

This research was supported by a grant (19SCIP-C116873-04) from the Construction Technology Research Program funded by the Ministry of Land, Infrastructure, and Transport of the Korean government.

REFERENCES

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

He, K. *et al.* (2016) 'Deep Residual Learning for Image Recognition', *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778. Available at: http://image-net.org/challenges/LSVRC/2015/ (Accessed: 17 September 2018).

He, K. *et al.* (2017) 'Mask R-CNN', *Proceedings of the IEEE International Conference on Computer Vision*, 2017-Octob, 2980–2988. doi: 10.1109/ICCV.2017.322.

Hutchinson, T. C. and Chen, Z. (2010) 'Image-based framework for concrete surface crack monitoring and quantification', *Advances in Civil Engineering*, 2010. doi: 10.1155/2010/215295.

Kim, B. and Cho, S. (2018) 'Automated Vision-Based Detection of Cracks on Concrete Surfaces Using a Deep Learning Technique', *Sensors*, 18(10), 3452. doi: 10.3390/s18103452.

Kim, B. and Cho, S. (2019) "Image-based Concrete Crack Assessment using Mask and Region-based Convolutional Neural Network," Structural Control and Health Monitoring, doi: 10.1002/stc.2381.

Lin, T. Y. *et al.* (2014) 'Microsoft COCO: Common objects in context', in. doi: 10.1007/978-3-319-10602-1_48.

Long, J., Shelhamer, E. and Darrell, T. (2015) 'Fully convolutional networks for semantic segmentation', *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June, 3431–3440. doi: 10.1109/CVPR.2015.7298965.

Ren, S. *et al.* (2017) 'Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. doi: 10.1109/TPAMI.2016.2577031.

Valeti, B. and Pakzad, S. (2017) 'Automated Detection of Corrosion Damage in Power Transmission Lattice Towers Using Image Processing', in *Structures Congress 2017*. Reston, VA: American Society of Civil Engineers, 474–482. doi: 10.1061/9780784480427.040.

Yang, E. *et al.* (2018) 'Automated Pixel-Level Pavement Crack Detection on 3D Asphalt Surfaces with a Recurrent Neural Network', *Computer-Aided Civil and Infrastructure Engineering*, 34, 213–229. doi: 10.1111/mice.12409.