

Autonomous Bolt Loosening Detection using Deep Learning

Yang Zhang^{a,b}, Xiaowei Sun^c, Kenneth J. Loh^d, Wensheng Su^c, Zhigang Xue^c, Xuefeng Zhao^{a,b,*}

^a School of Civil Engineering, Dalian University of Technology, 116024, Dalian, China;

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

^c Jiangsu Province Special Equipment Safety Supervision Inspection Institute, Branch of Wuxi, 214174, Wuxi, China;

^d Department of Structural Engineering, University of California-San Diego, La Jolla, CA 92093, USA.

* Corresponding author e-mail: zhaoxf@dlut.edu.cn

ABSTRACT

Machine vision-based structure health monitoring (SHM) is gaining popularity due to the rich information one can extract from video and images. However, the extraction of characteristic parameters from images often requires manual intervention, thereby limiting its scalability and effectiveness. In contrast, deep learning overcomes the aforementioned shortcoming in that it can autonomously extract feature parameters (*e.g.*, structural damage) from image datasets. Therefore, this study aims to validate the use of machine vision and deep learning for SHM by focusing on a particular application of detecting bolt loosening. First, a dataset that contains 200 images was collected. The dataset includes two bolt looseness states, namely, tight and loosened. Second, a faster regional convolution neural network (Faster R-CNN) was trained and evaluated. The test results showed that the average precision of bolt damage detection is 0.9503. Thereafter, bolts were loosened to various screw heights, and images obtained from different angles and lighting conditions were identified separately. The trained model was then employed to validate that bolt loosening could be detected with sufficient accuracy using various types of images. Finally, the trained model was connected with a webcam to realize real-time bolt loosening damage monitoring.

Keywords: bolt loosening, convolutional neural network, deep learning, imaging, machine vision

1. Introduction

Aerospace, civil, marine, and mechanical structures can sustain damage any time during their entire operational life cycle. If damage is left undetected, it can propagate to cause component or catastrophic structural failure. Therefore, the aim of any structural health monitoring (SHM) system is to detect early stages of structural damage so that appropriate repairs can be conducted to maintain system performance and functionality. In addition to confirming whether damage exists or not, the goal is to also characterize severity, identify the location, and differentiate the types of damage in the system.

Conventional SHM methods mainly rely on measuring structural response due to ambient or forced excitations, typically using sensors such as accelerometers, strain gages, displacement meters, and inclinometers. Then, modal analysis is used to determine the degree of structural damage [1], such as by using dynamic fingerprint analysis, system identification, and neural networks [2]. These methods are all based on characterizing the dynamic properties of the structure, which is highly dependent on

environmental effects and insensitive to local damage. Furthermore, these detection methods only suitable for simple (or idealistic) structures. Although the deployment of dense sensor networks (*e.g.*, wireless sensors) could improve structural monitoring performance, structural dynamic properties do not change significantly due to highly localized damage features (*e.g.*, bolt loosening or fatigue crack).

With the advancement of cameras and other image capturing technologies, machine-vision-based damage detection methods have developed rapidly, simply because machine vision can be adapted to diverse and complex engineering environments with high precision and high speed. A common approach is to feed image data streams to image processing algorithms for extracting abnormal features indicative of damage or change relative to a healthy baseline. Currently, machine vision has been applied in many fields, such as bridge monitoring [3-5], vehicle monitoring [6, 7], dam monitoring [8], and structural cracking detection [9-11]. Overall, vision-based damage detection methods are intuitive and effective at monitoring structural cracks, corrosion and other common types of damage.

In recent years, with the rapid developments of deep learning, various algorithms have emerged with the ability to classify and recognize structural features at high accuracy. This enables automation and eliminates the need to manually process and compare processed image results with an assumed, constant, baseline. In fact, to date, deep learning has been widely used in medical diagnosis [12-15], data mining [16], autopilot [17-20], and literary translation [21-22], which have demonstrated remarkable results and success. In general, all of the deep learning fields are mainly divided into three domains: (1) image classification; (2) speech recognition; and (3) text understanding. In particular, significant achievements have been made in image classification largely due to the remarkable performance of convolutional neural networks (CNN). For example, in 2017, Cha et al. [23] proposed a crack recognition method based on CNN and introduced deep learning to the field of structural health monitoring for the first time. Their results showed that this method can meet the engineering needs and has high recognition accuracy. Therefore, combining deep learning with machine vision can be readily applicable for damage identification in civil engineering structures.

One of the most important structural components in buildings is its connections, where bolt damage can adversely affect structural safety and performance. Currently, many researchers have used machine vision to study the problem of damage in the form of bolt loosening. For example, Hough transform was used to identify the bolt edge, and the degree of bolt damage was judged by identifying the rotation angle of the bolt edge [24]. The length of the bolt was also assessed by support vector machine to determine the degree of bolt looseness [25, 26]. While all of these methods have high accuracy, they all need to artificially extract damage characteristics through many intermediary steps. In addition, this procedure suffers from greater subjectivity when extracting damage characteristics. Thus, it is very difficult to actualize automated end-to-end damage monitoring using these aforementioned techniques. Based on the present situation, this paper proposes a new bolt looseness damage detection method using deep learning.

This paper combines machine vision and deep learning to propose a new monitoring method that can realize automated, end-to-end, bolt looseness damage detection without having to rely on feature extraction. First, a test structure was designed, and image datasets were collected using a typical smart phone. The dataset was separated such that 64% of it was used for training, 16% for validation, and 20% for testing. Then, a faster region based convolution neural network (Faster R-CNN) algorithm was trained and evaluated using the aforementioned image datasets. Second, this study then examined its applicability for damage detection by considering the extension of the bolt as a potential damage state of interest. In order to verify the accuracy of the new method, images of different bolts tightened to various

heights and angles were identified. Third, to further introduce realism, the effect of lighting conditions on recognition accuracy of the detection method was analyzed. It was found that images that was taken in poorly lit conditions made damage detection more challenging; nevertheless, this method still showed strong detection capabilities under different lighting conditions. Then, to validate this method, four images of loosened bolts that were acquired from other structures were successfully detected by the trained model. Finally, in order to meet the needs of real-time monitoring, the trained model was connected with a webcam to detect and locate bolt looseness damage status. In summary, the new method not only enabled high precision and accurate damage detection, but it also met the requirement of real-time monitoring, which is a necessary step for realizing automated, end-to-end, damage monitoring.

2. Experimental Details

2.1 Experimental Test Structure

This paper proposes a new detection method that combines deep learning with machine vision to identify and locate bolt looseness damage. In this method, looseness of the bolt was defined by the extension length of the screw (or threaded bolt). Two states were defined: tight (intact) and loose (damaged). Figure 1 shows the test structure employed, where multiple bolts of different states could be introduced simultaneously. The test structure consisted of three iron plates, in which two plates were used as the supports, and the other large, horizontal plate was used for installing nine M28 bolts. The dimensions of the plate was 400 mm long, 240 mm wide, and 10 mm thick. The spacing between bolts was 70 mm. Then, the Faster R-CNN algorithm was trained and assessed using image datasets obtained from this test setup. Thereafter, the trained model was used to detect the bolt looseness or damage.

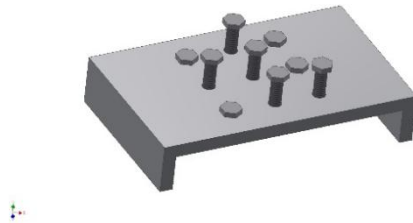


Figure 1. Schematic diagram of experiment structure

2.2 Image Datasets

The image datasets of the test structure that were used in this article were collected as shown in Figure 2. Initially, when a bolt is tightened, the length of the extend screw was 0 cm. Then, by loosening the bolt, the length of the extended portion of the screw was ~ 3 cm. Smart phones were used to acquire images of the bolts and test structure, and these images were taken at various angles and distances. The specifications of the onboard camera of smartphone are shown in Table 1. A total of 200 photos were acquired (4032×3016 pixels). In order to increase the training speed of the deep learning algorithm, all of the pictures were uniformly converted to 640×478 pixels. The datasets were then separated into 64% for training (128 images), 16% for validation (32 images), and 20% for testing (40 images). In addition, a tagging tool was used to tag all images as shown in Figure 2.

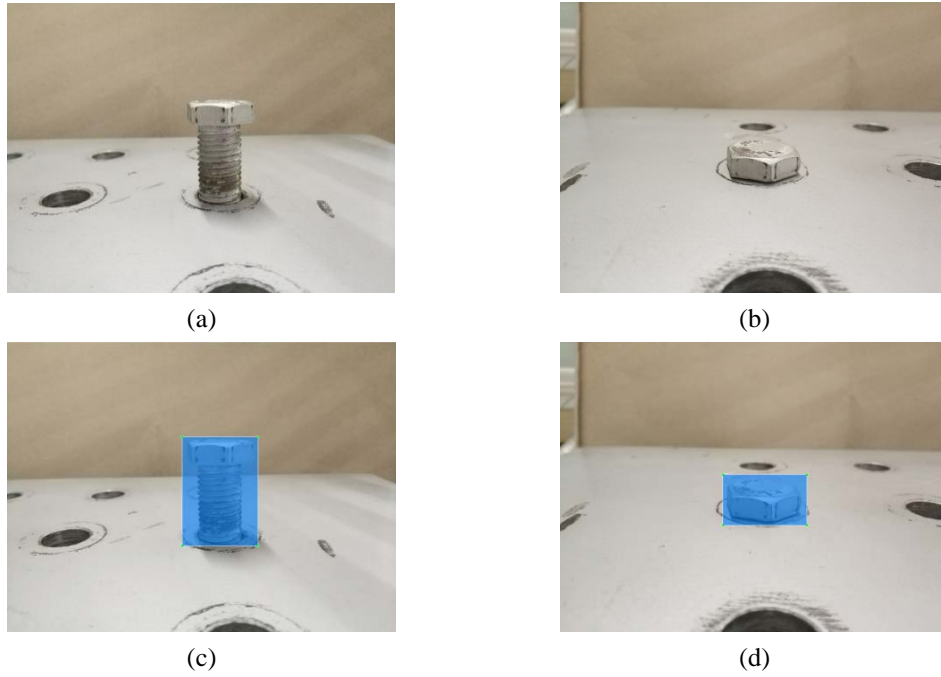


Figure 2. (a) A loosened and (b) tightened bolts, as well as the corresponding tagged images for the (c) damaged and (d) intact bolts, are shown.

Table 1. Smart phone camera specifications

Parameters	Value
Size	4032 × 3016 pixels
Vertical resolution	72 dpi
Horizontal resolution	72 dpi
Bit depth	24
Aperture	f/1.8

2.3 Hardware and Software Configurations

The training of deep learning networks requires ample computing power. Thus, a Dell 7810 workstation was used for training the model. The workstation featured NVIDIA Geforce GTX 1080 Ti (with 11 GB of memory) and ran software such as the Ubuntu system, CUDA8.0, cuDNN6.0, python3.6, and tensorflow1.4, among others. The deep learning network used in this study is Faster R-CNN, as mentioned previously, and the convolutional neural network framework is VGG16.

3. Faster R-CNN

The central hypothesis of this study is that convolutional neural networks can be employed to extract structural damage features, such as bolt loosening, from image datasets. However, the problem of image positioning needs to be solved first. In the past, sliding windows were mostly used to divide an intact picture into multiple smaller pictures. Then these small pictures were classified and recognized for image positioning, which can be inefficient. Instead, locating objects can be achieved by integrating an object detection algorithm with CNN. There are two main approaches to implement this. The first is to treat it as a regression problem, and the second is to use the anchor boxes. The Faster R-CNN used in this study belongs to the latter. Faster R-CNN can not only accurately find the location of an object, but it can also identify the category of the object [27-29]. The overall framework of the Faster R-CNN is shown in

Figure 3.

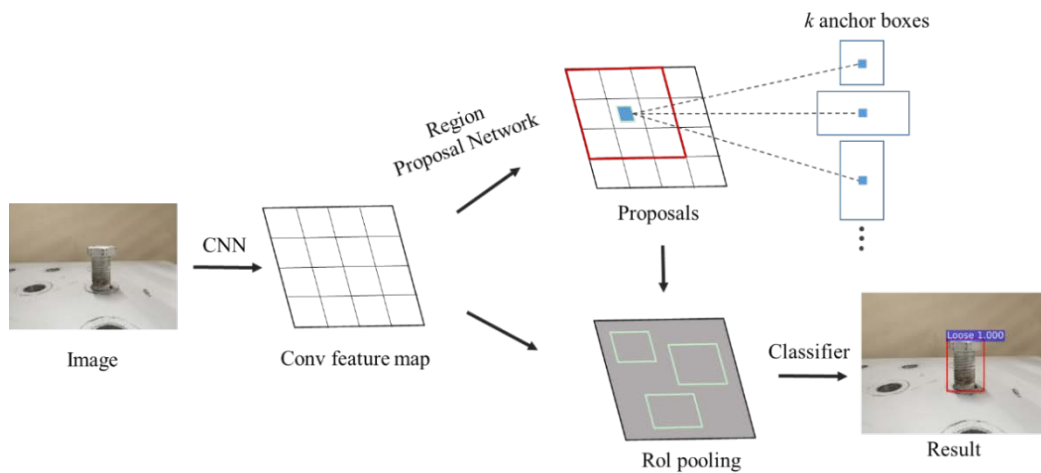


Figure 3. The Faster R-CNN framework

3.1 Convolutional Neural Networks

Deep learning extracts the features of pictures using a convolutional neural network. In particular, Faster R-CNN uses the Convolutional layers, ReLU layers, and Pooling layers to extract feature maps of images. This feature map is used by region proposal networks and the fully connected layer. In Faster R-CNN, all of the convolutional layers are expanded to convert the original image size of $M \times N$ to $(M+2) \times (N+2)$, and the output is $M \times N$ after the convolution of 3×3 . This process is shown in Figure 4. Similarly, the matrix will change to $(M/2) \times (N/2)$ after each pooling layer. Thus, the convolutional layer does not change the sizes of the input and output in this convolutional network; however, the pooling layer changes both the length and width of the output to become half of the input. Therefore, the size of the matrix changes from $M \times N$ to $(M/16) \times (N/16)$ when it is processed by the convolutional section. Because the feature map is defined by the matrix, there is direct correspondence between the feature map and the original image.

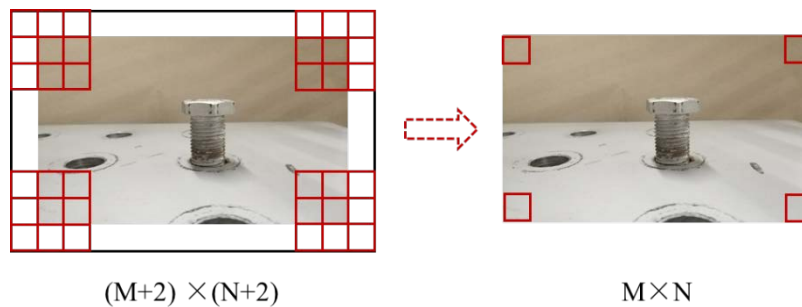


Figure 4. The convolution process after the original image is expanded

3.2 Region Proposal Networks

In the past, the speed of object detection algorithms that generated region proposals are very slow, such as the case for sliding windows and selective search. The Faster R-CNN uses region proposal networks to generate rectangular object proposals, which greatly improves the generation speed of region proposals. The region proposal network is illustrated in Figure 5. First, Faster R-CNN will reset any size of image to 800×600 pixels, followed by generating nine rectangular object proposals corresponding to the size of the reset image, including three ratios 1:1, 1:2, and 2:1. These rectangular object proposals basically cover the various dimensions and shapes of the reset image. Each point on the feature map will match

those of the nine detection boxes, although it should be noted that these original detection boxes are inaccurate. In addition, there are also two bounding box regressions to correct the position of the detection boxes. More importantly, this method can share convolutional features with the region proposal and object detection. Then, the fully connected layer and Softmax layer are used to calculate the category of each region. At the same time, bounding box regression is used to calculate the offset of each object proposal position for obtaining a more accurate object detection proposal.

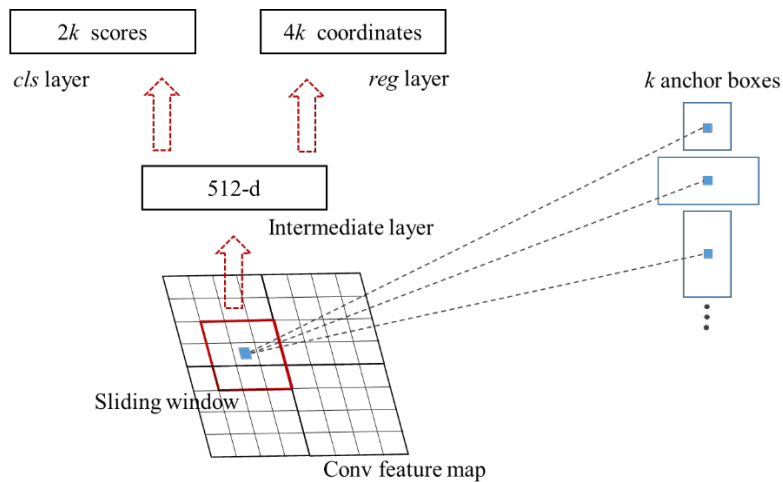


Figure 5. Region Proposal Network

3.3 VGG16

The network framework used in this study is VGG16, which includes 13 Convolutional layers, 13 ReLU layers, 5 Max pooling layers, 3 Fully connected layers, and 1 Softmax layer, as shown in Figure 6. For all the convolutional layers, kernel size is 3, and pad is 1. For all pooling layers, kernel size is 2, and stride is 2. The size of the input image is 224×224 pixels. The kernel size of 3 increases the ability of nonlinear expression, thereby making the segmentation plane more separable. VGG16 uses min-batch gradient descent to optimize multinomial logistic leans.

Faster R-CNN was trained on a model that was already well-trained, such as VGG, ZF, and Res. The network model used in this article is VGG16, as was mentioned earlier and shown in Figure 6. The training process of Faster R-CNN was as follows: (1) training the RPN on the training model, where this step collected a series of region proposals; (2) training the Fast R-CNN; (3) training RPN next, where this section collected a series of region proposals again; and (4) training Fast RCNN for the second time. This training process repeated a total of two times. The reason of only repeating two times was that the training effect was not greatly improved as the number of repetitions was increased further.

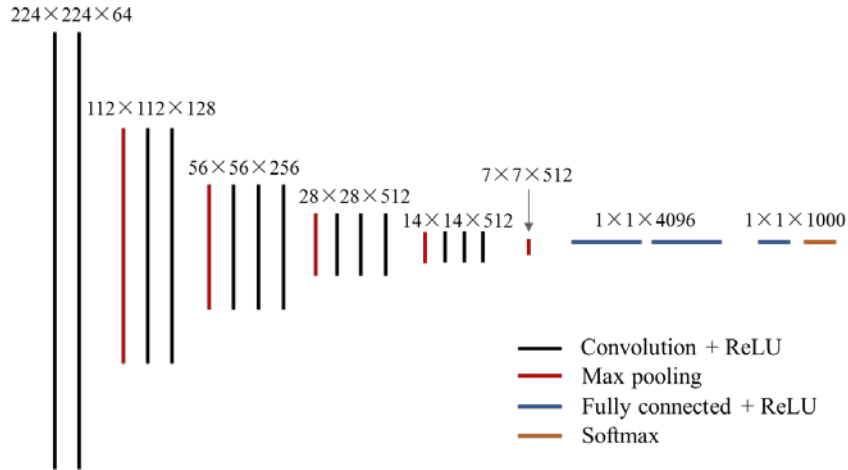


Figure 6. Microarchitecture of VGG16

4. Bolt Damage Object Detection based on Deep Learning

4.1 Training Process

In this study, Faster R-CNN based on TensorFlow frame was used to identify and locate bolt looseness damage. The training process with a learning rate of 0.001, momentum of 0.9, and weight decay of 0.0005 for 3,000, 5,000 and 10,000 iterations was utilized. The aforementioned parameters were shared with the region proposal networks and object detection algorithm. Then, the anchor scales for RPN were 8, 16, and 32, while the anchor ratios for RPN were 0.5 and 1.2. The evaluation parameter of IOU was 0.3. In order to ensure convergence of the training results, three iterations were attempted to train the datasets. The training results are shown in Table 2, and total loss is plotted with respect to the total number of iterations as shown in Figure 7.

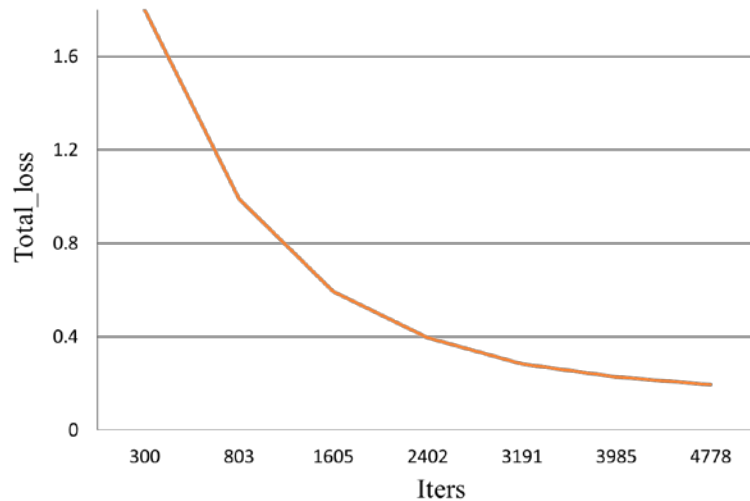


Figure 7. Total loss after 5,000 iterations

Table 2. Average precision (AP) corresponding to different iterations

Iterations	AP for Tight	AP for Loose	mAP
3,000	0.9100	0.9091	0.9095
5,000	0.9005	1.0000	0.9503
10,000	0.8559	0.9960	0.9260

When the number of iterations reached 5,000, the training result successfully converged, as shown in Figure 7. The average precision results are summarized in Table 2, where the mean average precision (mAP) was found to be 0.9503, which is acceptable for engineering applications. The mean average precision for 10,000 iterations is 0.9260, which is smaller than the case for 5,000 iterations. Based on these results, it was decided that 5,000 iterations were used for the model.

4.2 Bolt Looseness Damage Detection under Different Conditions

(1) Minimum resolution for bolt looseness detection

It was mentioned earlier that loosened bolts were defined as those with a screw height of 3 cm. However, the model which was trained by deep learning has the potential to generalize the definition of loosening, which can therefore be used to identify loosened bolts with different screw heights. Thus, in this section, a series of images with different screw heights were tested for finding the minimum resolution for bolt looseness detection.

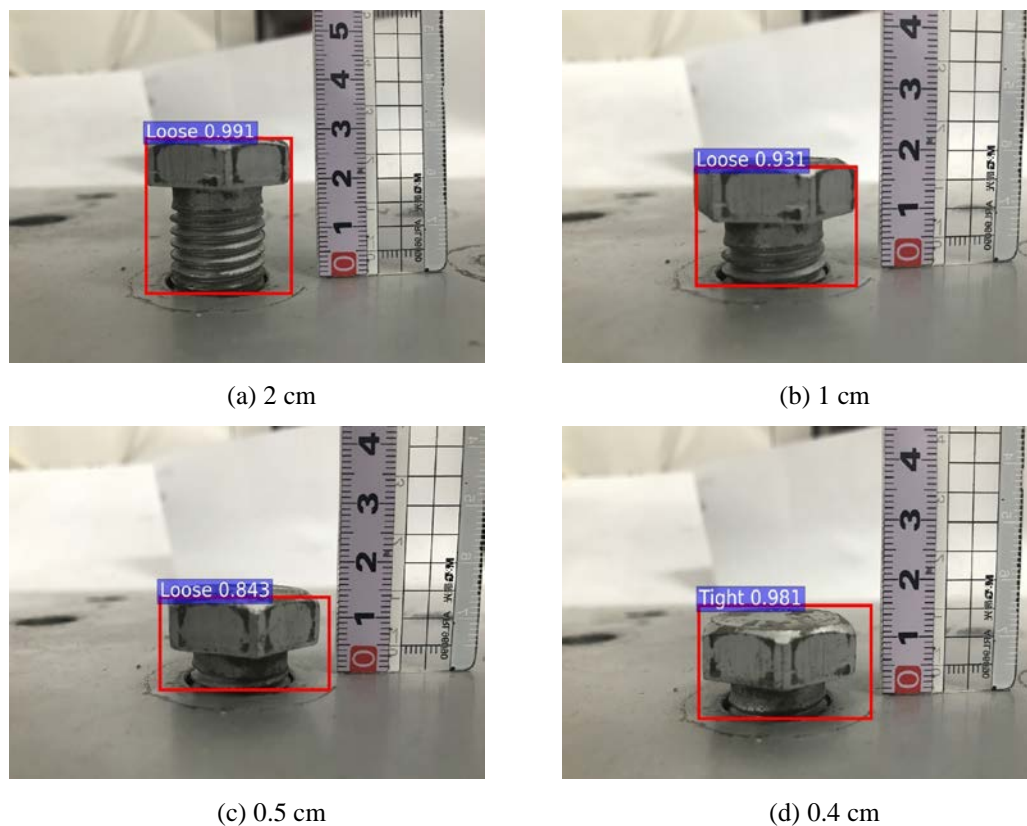


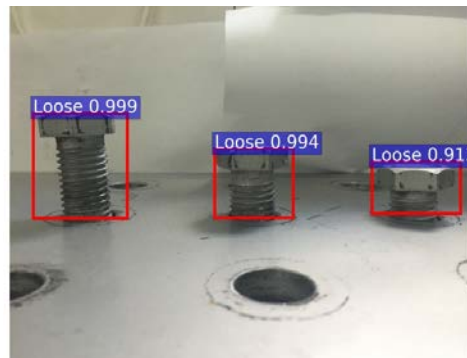
Figure 8. The detection result of different screw heights

Representative test results are shown in Figure 8. The four images with screw heights of 2 cm, 1 cm, 0.5 cm, and 0.4 cm were identified in Figure 8. Among them, the loosened bolts with screw heights of 2 cm, 1 cm, and 0.5 cm were correctly identified as “Loose”, and it was found that their recognition accuracy decreased as the height decreases (which makes sense, as these became more difficult to recognize visually). Therefore, the screw heights of bolts can affect bolt looseness recognition. On the other hand, the loosened bolt with a screw height of just 0.4 cm was mistakenly identified as “Tight”. Based on test results obtained in this study, it can be concluded that the minimum resolution for bolt looseness detection was 0.5 cm, based on the parameters of testing. Nevertheless, even though the detection method was trained with screw heights of 3 cm, the technique was successfully generalized to detect loosened bolts of different screw heights up to a minimum of 0.5 cm. These results also indicated that the recognition

effect of this method was not limited to just the characteristics of the datasets and possessed a strong learning ability.

(2) Looseness detection under different angles

In order to verify that the training model could identify bolt looseness damage based on images acquired at different angles, bolts of different loosened heights (*i.e.*, 3 cm, 2 cm, and 1 cm) were photographed at a low angle (0°), medium angle (0° to 45°), and high angle (45° to 90°), as shown in Figure 9. Then, these images were used to see if the Faster R-CNN framework could identify that these bolts were in fact loose.



(a) Low angle



(b) Medium angle



(c) High angle

Figure 9. The detection result of different angles

The test results are shown in Figure 9. The images acquired at low and medium angles successfully identified that all three bolts were loose. On the contrary, for the high angle image, the bolt with a loosened height of 1 cm was incorrectly identified as being tight. These results are important, since it clearly shows that detection methods based on machine vision is often restricted by camera shooting angle, which can lead to some errors during damage detection and is unavoidable. Overall, the precision

of bolt damage detection based on this method is acceptable.

(3) Looseness detection under different lighting conditions

In order to detect the stability of the training model, images taken under different lighting conditions were evaluated. These pictures are divided into two categories, namely, normal lighting and dark lighting. Using the image datasets, the individual bolts were identified, and the training results showed that the framework could locate and identify looseness damage of multiple bolts simultaneously. The looseness damage of nine bolts on the test structure was obtained from one image, where the image size was $3,264 \times 2,448$ Pixel. Each image was tested separately, and the average time required for each image was 0.280 s. Some representative recognition results are shown in Figure 9. In addition, the statistics of the test results are summarized in Table 3.

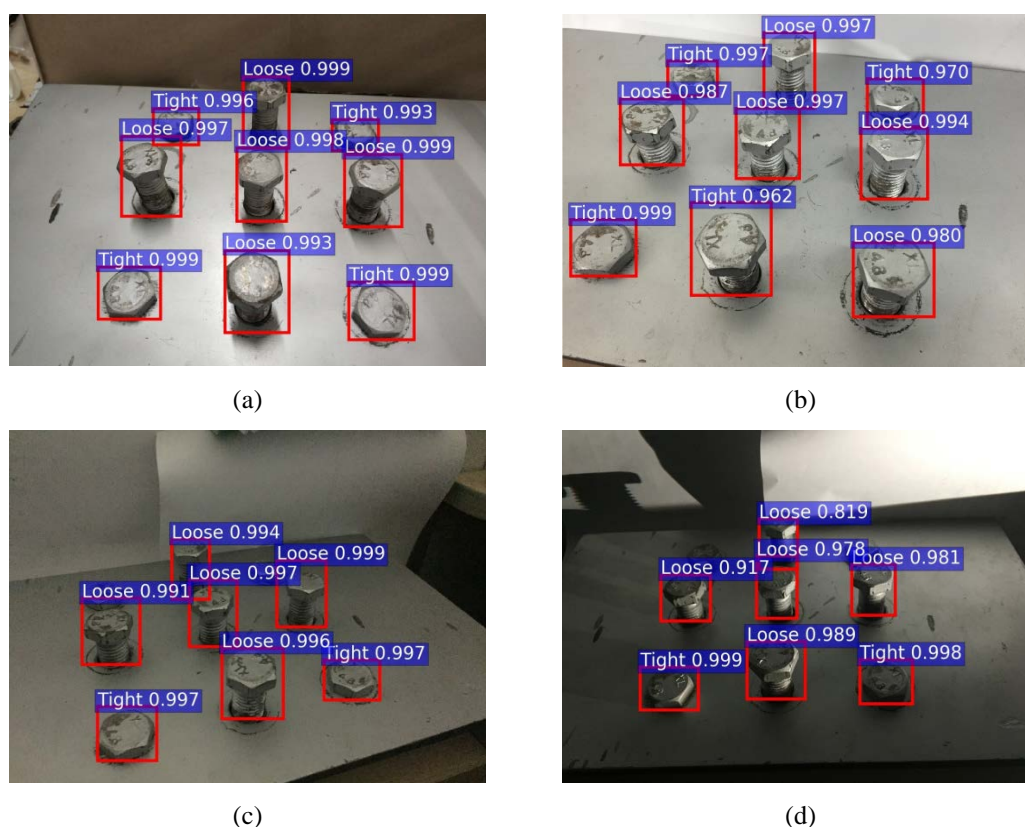


Figure 10. The detection results of bolt looseness under different lighting conditions

As can be seen from Figure 10, the recognition accuracy of this method was still high under normal and dark lighting conditions. Table 3 computes statistical parameters that quantifies recognition performance. In Table 3, P is positive, N is negative, TP is true positive, TN is true negative, FP is false positive, and FN is false negative. The identification accuracy, prediction accuracy, and recall rate for all kinds of cases encountered are listed in Table 3. The recall rate of all four images was 100%; the precision of three images are 100%, while the other image had a damage detection error of one bolt, mainly because of the effect of shooting angle. In addition, as a result of the dark lighting condition, two bolts in the back row were not clearly photographed, so they were not identified. This caused the recognition accuracy to be 77.78% in Figures 10c and 10d. Therefore, based on these results, poor lighting had a degree of influence on recognition precision of the framework, but it still satisfied the requirement of identifying and locating the bolts in the test structure. This experiment showed that the sharpness of the image had an important influence on recognition results, but shadow

and poor lighting did not adversely affect recognition accuracy.

Table 3. The detailed analysis of damage detection results based on different lighting conditions

Test sample	P	N	TP	TN	FP	FN	Accuracy	Precision	Recall
10a	5	4	5	4	0	0	100%	100%	100%
10b	5	4	4	4	1	0	88.89%	80%	100%
10c	5	4	5	2	0	0	77.78%	100%	100%
10d	5	4	5	2	0	0	77.78%	100%	100%

(4) Bolt looseness detection on other structures

In order to prove that the model could be generalized and used for different applications (even though training was based on bolts on the experimental test structure), images of loosened bolts were obtained from the two other completely different structures for validation. The detection results are shown in Figure 11. Because the bolts used for model training and those on the other structures are not of the same type, the localization accuracy of this method was inferior as compared to that of the previous section; nevertheless, high recognition accuracy was achieved. To further enhance bolt and damage detection capabilities, one could introduce different types of bolts to the training dataset in the future.

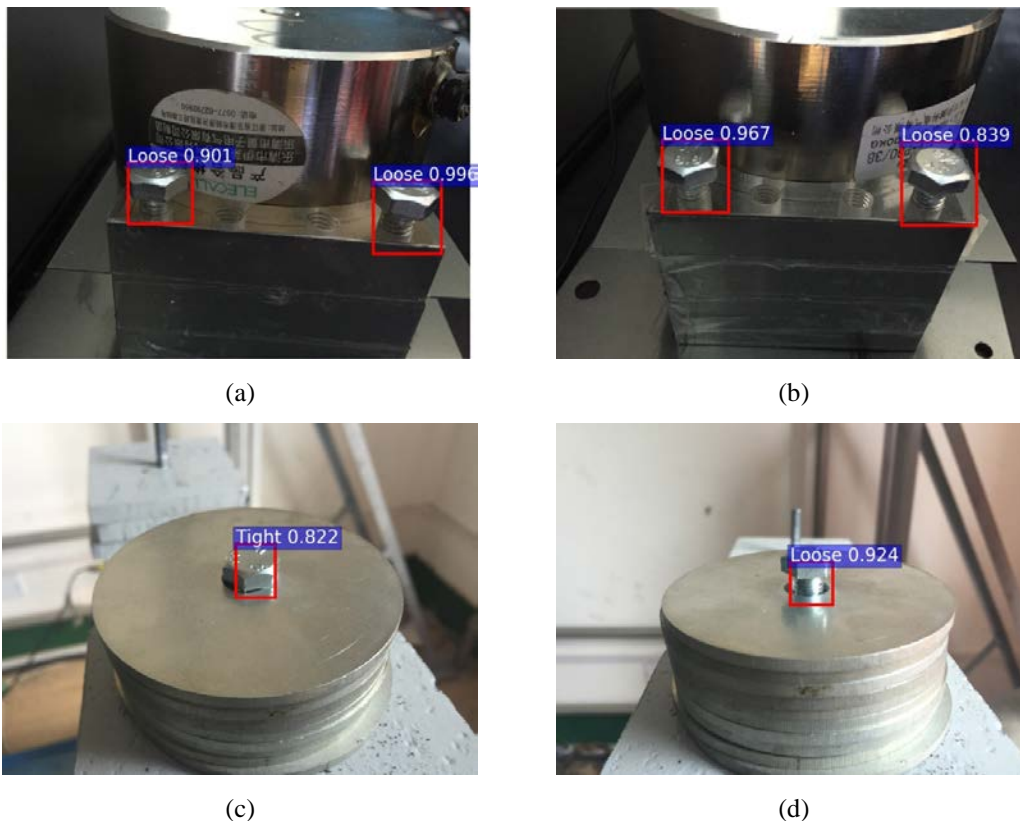


Figure 11. Bolt detection validation results when bolts were installed on different structures

5. Real-time Bolt Looseness Monitoring using a Webcam

The above parts are the bolt looseness damage detection based on image. First, the data sets were used to complete the model training. Second, the bolt damage images under different conditions were collected. And then these images were copied to the computer for testing by the training model. Obviously, the whole process is not continuous, it cannot meet the requirements of real-time monitoring. So the webcam is connected with the deep learning network in this section. The training model was used to directly

recognize the images collected by the webcam in real time, this process as shown in the Figure 11. This experiment adopts USB industrial camera, some parameters of lens are listed: M1214-MP2, focal length 12mm, maximum imaging size 8.8×6.6 ($\Phi 11$), aperture range F1.4~F16C, working distance 0.15~inf(m), filter thread M30 .5 \times P0.5, mechanical dimensions $\Phi 33.5 \times 28.2$.

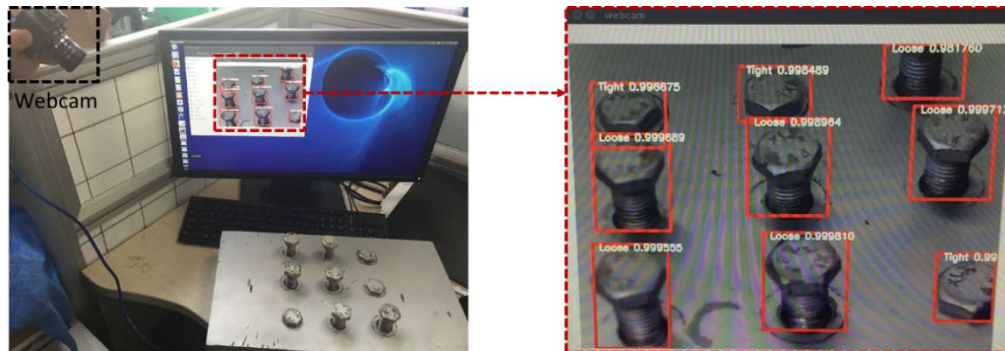


Figure 12. Bolt looseness real-time monitoring

It can be seen from Figure 12 that the status of the nine bolts were collected in real time using the webcam, and the deep learning model can directly locate and identify the bolt looseness damage of the images captured by the camera. Six of the nine bolts on the experimental structure were in the looseness state and the others were in the tightened state. The bolt damage identification method based on deep learning can accurately identify and locate the nine bolts damage, and the identification probability of each bolt state is greater than 0.98. It shows that the detection method not only has high recognition accuracy, but also can meet the online recognition using the webcam. Therefore, a bolt looseness real-time monitoring system can be built, it can meet the actual needs of the engineering to the greatest extent and provide the early damage warning for the safety of the engineering.

6. Conclusion

This paper proposes a new bolt looseness damage monitoring method based on deep learning. The Faster R-CNN was used to train the data sets, the network doesn't require a large data sets to get a good recognition effect, greatly reducing the difficulty of data sets collection. At the same time, the network is based on the identification algorithm of the region proposals, which can complete the object detection and location in a network. The test results show that the training model has high recognition accuracy and the average precision can reach 0.9503. Although the screw length of the all loosened bolt in the data sets are 3cm, but the training model can still accurately identify the bolt looseness damage with a screw length of 0.5cm. However, the recognition accuracy is limited by the shooting angle. The recognition results under different angles of view may also be different. This is a common problem of machine vision and can be solved by rotating the camera to reduce the possibility of misclassification. The bolt looseness damage images taken in different environments and different structures have excellent recognition effects, indicating that the training model has strong generalization ability and robustness. Finally, in order to meet the requirements of the damage real-time monitoring in the engineering, the deep learning is connected with the webcam, which can identify and locate the damage that appears in the webcam's visual field in real time. In short, the bolt looseness damage monitoring technology based on deep learning has high recognition accuracy, and can realize the real-time monitoring of damage, greatly improving the ability of practical application in engineering.

Acknowledgments

This research was supported by the Science and Technology Project of China Jiangsu Province Special Equipment Safety Supervision Inspection Institute in 2016 (KJ (Y) 2016008), National Natural Science Foundation of China (51479031), and the National Key R&D Program of China during the Thirteenth Five-Year Plan Period (2016YFC0802002). Prof. Loh was supported by the Jacobs School of Engineering, University of California-San Diego.

References

- [1] Doebling, S. W. "Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics : A Literature Review." *Shock & Vibration Digest* 30.11(1996):2043-2049.
- [2] Housner, G. W., et al. "Structural Control: Past, Present, and Future." *Journal of Engineering Mechanics* 123.9(1997):897-971.
- [3] Oh, Je Keun, et al. "Bridge inspection robot system with machine vision." *Automation in Construction* 18.7(2009):929-941.
- [4] Kim, S. W., et al. "Vision-based monitoring system for evaluating cable tensile forces on a cable-stayed bridge." *Structural Health Monitoring* 12.5-6(2013):440-456.
- [5] Luo, L., M. Q. Feng, and Z. Y. Wu. "Robust vision sensor for multi-point displacement monitoring of bridges in the field." *Engineering Structures* 163(2018):255-266.
- [6] Trivedi, Mohan Manubhai, Tarak Gandhi, and Joel McCall. "Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety." *IEEE Transactions on Intelligent Transportation Systems* 8.1 (2007): 108-120.
- [7] Zhu, Zhigang, et al. "VISATRAM: A real-time vision system for automatic traffic monitoring." *Image and Vision Computing* 18.10 (2000): 781-794.
- [8] Zaczek-Peplinska, Janina, et al. "Application of Digital Image Correlation and Geodetic Displacement Measuring Methods to Monitor Water Dam Behavior under Dynamic Load." *Journal of Civil Engineering and Architecture* 9.12 (2015): 1496-1505.
- [9] Jahanshahi, M. R., and S. F. Masri. "A Novel Vision-Based Crack Quantification Approach by Incorporating Depth Perception for Condition Assessment of Structures." *International Conference on Computing in Civil Engineering* 2012:531-536.
- [10] Shan, Baohua, S. Zheng, and J. Ou. "A stereovision-based crack width detection approach for concrete surface assessment." *Ksce Journal of Civil Engineering* 20.2(2016):803-812.
- [11] Jahanshahi, Mohammad R., and S. F. Masri. "Adaptive vision-based crack detection using 3D scene reconstruction for condition assessment of structures." *Automation in Construction* 22(2012):567-576.
- [12] Li, Rongjian, et al. "Deep Learning Based Imaging Data Completion for Improved Brain Disease Diagnosis." *Med Image Comput Comput Assist Interv* 17.Pt 3(2014):305-312.
- [13] Nie, Liqiang, et al. "Disease Inference from Health-Related Questions via Sparse Deep Learning." *IEEE Transactions on Knowledge & Data Engineering* 27.8(2015):2107-2119.
- [14] Liu, Siqi, et al. "Early diagnosis of Alzheimer's disease with deep learning." *IEEE, International Symposium on Biomedical Imaging* IEEE, 2014:1015-1018.
- [15] Hammerla, Nils Y, et al. "PD disease state assessment in naturalistic environments using deep learning." *Twenty-Ninth AAAI Conference on Artificial Intelligence* AAAI Press, 2015:1742-1748.
- [16] Wlodarczak, Peter, J. Soar, and M. Ally. "Multimedia data mining using deep learning." *Fifth*

- International Conference on Digital Information Processing and Communications* IEEE, 2015:190-196.
- [17] Liu, Meng, J. Niu, and X. Wang. "An autopilot system based on ROS distributed architecture and deep learning." *IEEE, International Conference on Industrial Informatics* IEEE, 2017:1229-1234.
- [18] Bojarski, Mariusz, et al. "End to End Learning for Self-Driving Cars." (2016).
- [19] Shalev-Shwartz, Shai, et al. "Long-term Planning by Short-term Prediction." (2016).
- [20] Santana, Eder, and G. Hotz. "Learning a Driving Simulator." (2016).
- [21] Castelvechi, Davide. "Deep learning boosts Google Translate tool." *Nature* (2016).
- [22] Faisal, M., and S. Manzoor. "Deep Learning for Lip Reading using Audio-Visual Information for Urdu Language." (2018).
- [23] Cha, Young Jin, et al. Vision-Based Concrete Crack Detection Using a Convolutional Neural Network. *Dynamics of Civil Structures*, Volume 2. 2017.
- [24] Park, Jae Hyung, et al. "Vision-based technique for bolt-loosening detection in wind turbine tower." *Wind & Structures An International Journal* 21.6(2015):709-726.
- [25] Ramana, Lovedeep, W. Choi, and Y. J. Cha. "Fully automated vision-based loosened bolt detection using the Viola-Jones algorithm." *Structural Health Monitoring* 1(2018): 147592171875745.
- [26] Ramana, Lovedeep, W. Choi, and Y. J. Cha. "Automated Vision-Based Loosened Bolt Detection Using the Cascade Detector." (2017).
- [27] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.
- [28] Girshick, Ross. "Fast r-cnn." *arXiv preprint arXiv:1504.08083* (2015).
- [29] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems*. 2015.