

AUTOMATIC DETECTION OF TIMBER-CRACKS IN WOODEN ARCHITECTURAL HERITAGE USING YOLOv3 ALGORITHM

Yinghua Liu^{1,2}, Miaole Hou^{1,2,3}, Aiqun Li⁴, Youqiang Dong^{1,2,3*}, Linlin Xie⁴, Yuhang Ji^{1,2}

¹School of Geomatics and Urban Spatial Informatics, Beijing University of Civil Engineering and Architecture, No.15Yongyuan Road, Daxing District, Beijing – (Yinghua Liu, Miaole Hou, Youqiang Dong, Yuhang Ji) lyhbucea@163.com

² Beijing Key Laboratory for Architectural Heritage Fine Reconstruction & Health Monitoring, No.15Yongyuan Road, Daxing District, Beijing – (Miaole Hou, Yinghua Liu, Youqiang Dong, Yuhang Ji) houmiaole@bucea.edu.cn

³ Engineering Research Center of Representative Building and Architectural Heritage Database, Ministry of Education, No.15Yongyuan Road, Daxing District, Beijing – (Youqiang Dong, Miaole Hou) dongyouqiang@bucea.edu.cn

⁴School of Civil and Transportation Engineering, Beijing University of Civil Engineering and Architecture, No.15Yongyuan Road, Daxing District, Beijing – (Aiqun Li, Linlin Xie) liaiqun@bucea.edu.cn

Commission II, WG II/8

KEY WORDS: Wooden architectural heritage, Timber-cracks, Detection, YOLOv3 algorithm, DarkNet-53, Dataset

ABSTRACT:

As there usually exist widespread crack, decay, deformation and other damages in the wooden architectural heritage (WAH). It is of great significance to detect the damages automatically and rapidly in order to grasp the status for daily repairs. Traditional methods use artificial feature-driven point clouds and image processing technology for object detection. With the development of big data and GPU computing performance, data-driven deep learning technology has been widely used for monitoring WAH. Deep learning technology is more accurate, faster, and more robust than traditional methods. In this paper, we conducted a case study to detect timber-crack damages in WAH, and selected the YOLOv3 algorithm with DarkNet-53 as the backbone network in the deep learning technology according to the characteristics of the crack. A large timber-crack dataset was first constructed, based on which the timber-crack detection model was trained and tested. The results were analyzed both qualitatively and quantitatively, showing that our proposed method was able to reach an accuracy of more than 90% through processing each image for less than 0.1s. The promising results illustrate the validity of our self-constructed dataset as well as the reliability of YOLOv3 algorithm for the crack detection of wooden heritage.

1. INTRODUCTION

Due to the unique natural and human factors, many wooden architectural heritages (WAH) were formed in the East (especially in East Asia). For instance, more than 70% of the frames of the ancient buildings in China are made of wood (Dai, Chang, Qian, Li, 2016), and more than 90% of the ancient buildings in Japan own wooden structure (Yang, 2016). WAH have extremely high historical, cultural and artistic value. Wood as a building material has many advantages, such as easy processing, but its usage is limited due to its own biological material characteristics. As time flies and environment changes, there appear many types of damage such as crack, decay, voids, and etc. Timber-cracks are the most typical damage type in WAH. Irreversible timber-cracks not only affect the appearance, but also reduce the load capacity of the components, reducing the overall safety performance of the building (Fu, 2016). Under the circumstances of widespread and large number of timber-cracks in WAH, how to accurately and quickly detect timber-cracks is regarded as the basis for the current status assessment and the important prerequisite of future repair plans. In the field of geomatics, according to different data sources, crack detection methods can be mainly divided into two categories: 3D point clouds-based method and 2D images-based method (Wen, 2019).

(1) 3D point clouds-based method

Point clouds data records the high-precision 3D information of the target surface. The crack detection usually relies on the features of manual selection and organization, such as the normal vector. The crack recognition is then realized based on the edge extraction algorithm and the point clouds segmentation algorithm. However, it cost a lot to purchase the standing or hand-held laser scanners to collect point clouds data. Due to the occlusion caused by the high internal height and the interspersed and stacked components inside the WAH, it is difficult to collect complete information about the target. In addition, scattered point clouds data lacking features such as texture brightness, make it more difficult to accurately detect small timber-cracks.

(2) 2D images-based method

Image data contains the features of the target surface texture and color and so on. The acquisition equipment is relatively cheap, and the use of digital cameras or mobile phones can meet the requirements. Crack detection methods based on image data can be divided into image processing technology and deep learning technology.

● Image processing technology

It consists of two steps including target feature extraction and classification. Feature extraction uses feature-driven methods to segment or enhance target through artificially selecting intuitive features. Target classification needs classifiers, such as SVM classifier. Since the view of a single image is narrow and the background is relatively single, the accuracy of the result highly depends on the image quality. Due to the poor lighting

* Corresponding author: Youqiang Dong, E-mail address: dongyouqiang@bucea.edu.cn

conditions inside WAH, the contrast of the images obtained in some locations is slightly lower, making it difficult to accurately detect all timber-cracks.

●Deep learning technology

It integrates feature learning with classification in one model (Luo, 2019). Through iteratively training the weights and biases of the neural network based on the training samples labelled with positions and classes, it can learn multi-level features autonomously. Then the testing samples are imported into the trained network to obtain the pictures that are marked with the location and the class of timber-cracks. The relevant indicators are finally selected to calculate the reliability of the model. Deep learning technology is driven by big data, and has higher requirements on computer hardware, especially GPU. It is a hot topic in the field of computer vision.

Comparing with the image processing technology, deep learning technology has higher-volume calculation and a heavier model, but it does not need feature organization and extraction. It can effectively overcome the inferior situation such as image weak light, which is more robust (Wen, 2019). The current deep learning-based crack detection method is mainly applied to detect concrete cracks in roads and bridges, having certain achievements, but it is rarely used in the architectural heritage, especially rarer used in the study of WAH. The accuracy of the detection results based on deep learning technology largely depends on the quality of the dataset (Ji, Shi, Meng, 2018). Currently open crack datasets are mainly concrete cracks, such as CFD dataset, but all countries have no open dataset of timber-cracks due to the special cultural status of WAH.

Taking the fact that large numbers of cracks in WAH need to be automatically detected into account, we completed two tasks in this work. (1) In cooperation with related cultural protection units, we collected internal images of WAH and constructed a large and accurate dataset of timber-crack through image annotation and data augmentation. (2) A timber-crack detection model is proposed based on YOLOv3 algorithm, where the structural parameters of the DarkNet-53 network were modified to make it suitable for timber-crack detection.

As the overall framework of this paper shown in Figure 1, the rest of the article is arranged as follows. Section 2 introduces the selected methods in detail, including YOLOv3 algorithm and DarkNet-53 network. Section 3 elaborates the process of constructing the timber-crack dataset and training the models. Section 4 tests the model, analyzes the results from both qualitative and quantitative perspectives. Section 5 summarizes the whole paper and draws conclusions.

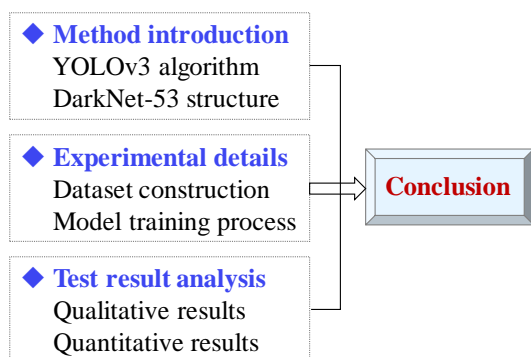


Figure 1 Main research content

2. METHODOLOGY

The deep learning technology-based target detection algorithms are divided into two-stage method and one-stage method. The two-stage method such as the RCNN algorithm, Fast RCNN algorithm, Faster RCNN algorithm, and etc., first uses the Region Proposal Network (RPN) to obtain the target candidate box for the image, which is automatically extracted and classified in the convolutional neural network. Then the candidate boxes containing the same target are merged to output the final detection result. This type of algorithm is highly accurate, but it is slow because of excessive computational cost. One-stage method such as YOLO algorithm, SSD algorithm, RetinaNet algorithm, and etc., does not use RPN to generate target candidate boxes. Instead, it directly returns the spatial position and class of the target in the final output layer. Thus, this kind of algorithm is faster, and it can also reach a higher level of accuracy through adopting deep convolutional neural network.

YOLO algorithm is one of the most widely used one-step detectors, proposed by Redmon et al. (2016). It has outstanding advantages such as fast speed and high precision. The improved version of YOLOv3 algorithm in 2018 better considers large and small targets (Ju, Luo, Wang, He, Chang, Hui, 2019). Its characteristics meet the needs of cultural relic protection for the rapid and accurate grasp of damages and it can well respond to damage detection of different sizes in WAH as well. The YOLOv3 algorithm has been widely used in the industrial field, but there is no application in the field of heritage protection. Therefore, this study uses the YOLOv3 algorithm for timber-crack detection.

2.1 Brief Introduction to YOLOv3

The idea of the YOLO algorithm is to split the original picture into small cells that do not coincide with each other, after which a feature map of this size is generated through a convolutional neural network, where each cell is used to predict the center point falling within the cell. The YOLO has undergone three versions. YOLOv1 changes the traditional sliding window operation to split the picture with a grid, separating detection target for each cell after segmentation. It uses LeakyRelu and GoogleNet as activation function and backbone network, respectively. Since it contains a full connection layer, that the input picture size needs to be fixed, resulting v1 pair of small target detection effect is poor (Redmon et al, 2016). YOLOv2 draws on the anchorbox mechanism in FasterRcnn and uses the K-means algorithm to cluster the dimension of the anchorbox. It uses BatchNormalization after each convolution layer and applies DarkNet-19 as backbone network. V2 removes the full connection layer, but the pooling layer still loses some features (Redmon, Farhadi, 2017). YOLOv3 draws on the ResNet residual structure to form a deeper network, borrows feature pyramid networks (FPN) upsampling and fusion methods to generate multi-scale feature maps for detecting targets at multiple scales, and uses backbone network for the full convolution of DarkNet-53 (Redmon, Farhadi, 2018). YOLOv3 algorithm mainly consists of three parts: pre-processing, convolutional operation, and logistic regression (see Figure 2).

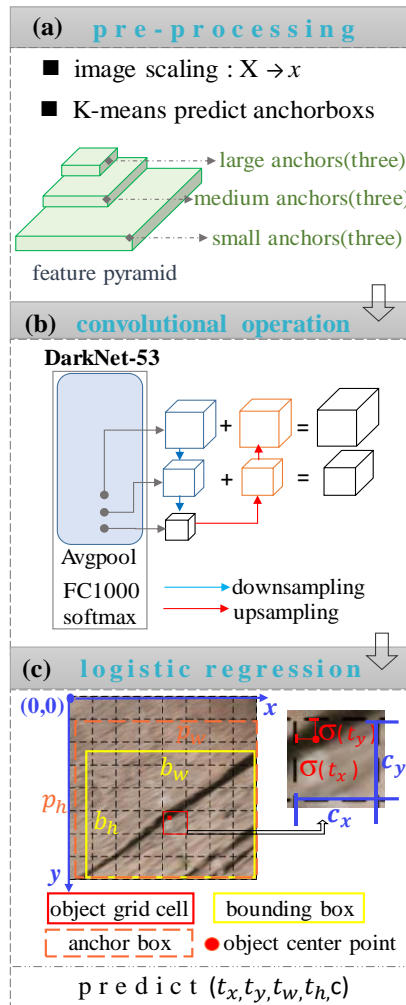


Figure 2. YOLOv3 algorithm flowchart

(A) Pre-processing: In order to accelerate network training and adapt computer hardware environment, the pictures need to be scaled. In addition, for many ground truths in the dataset, K-means clustering is used to obtain anchors of nine different scales, which are subsequently evenly distributed to feature maps of different sizes.

(B) Convolutional operation: Through using convolutional neural network based on the dataset to learning target features, the shallow network records the target fine-grained features well, while the deep network mainly learns the target semantic information. YOLOv3 algorithm makes two upsampling results of the last three subsamplings of the network. The same size feature map is stitched, which realizes the expansion of the tensor dimension. The network achieves the ability to learn the deep and shallow features of the target, finally producing three scale feature maps. As a result, the nine types of anchors generated in the previous step are evenly distributed. Among them, the feature map of the bigger receptive fields is allocated with a large ratio of anchors, while the feature map of the smaller receptive fields is allocated with a small ratio of anchors. The use of multi-scale feature map is conducive for detecting different size targets.

(C) Logistic regression: The YOLOv3 directly returns the spatial position and class probability of the target. Location prediction refer to predicting the boundingbox on each cell of the generated $S \times S$ size feature map. Each cell is assigned to three anchors, with the largest IOU to the ground truth

responsible for predicting the target, gradually approaching ground truth by fine-tuning this anchor (panning, scale zooming, etc). As shown in Figure 2(C), the regression position is to predict the (b_x, b_y) and b_w, b_h . Directly output the offsets t_x, t_y, t_w, t_h , and convert them by equation (1):

$$\begin{aligned} b_x &= \sigma(t_x) + c_x \\ b_y &= \sigma(t_y) + c_y \\ b_w &= p_w e^{t_w} \\ b_h &= p_h e^{t_h} \end{aligned} \quad (1)$$

where C_x, C_y are coordinates of the grid cell
 t_x, t_y are offset values of the target center point from the upper left corner of the grid cell
 σ refers to sigmoid function used to compress t_x, t_y to $[0,1]$
 p_w, p_h refer to width and height of the anchor mapped to the feature map
 b_x, b_y are center point coordinates of object
 b_w, b_h represents width and height of the boundingbox

Class prediction means using the softmax logic regression function to predict the probability of the classes to which the target belongs. It is an array of probabilities, the length of which is the total number of classes detected by the model. The function is:

$$\Pr(Class_i | Object) \quad (2)$$

where $Class_i$ refers the probability of each class to which the target belongs
Object refers the whether there is a target in boundingbox

Confidence prediction indicates the the confidence of the predicted boundingbox and groundtruth, where the mathematical representation of confidence is:

$$C = \Pr(Object) * IOU_{Object}^b \quad (3)$$

where IOU represents the ratio of the intersection of boundingbox and groundtruth
 b is the area of overlap, and object is the area of union

YOLOv3's loss function is also composed of the above three parts (i.e., localization, class, confidence), as shown in equation (4). Localization error refers to the error of the center coordinates and width and height of the boundingbox, represented by the mean squared loss function. Class error calculation uses binary cross entropy. Confidence error refers to the error rate between the presence and absence of targets in the boundingbox, which is calculated by binary cross entropy. The YOLOv3 target detection process is done in a neural network. The reverse propagation of the loss function can be carried throughout the network, optimizing the target detection performance by end to end.

$$L = Error_{localization} + Error_{class} + Error_{confidence} \quad (4)$$

where L represents YOLOv3's loss function
 $Error_{localization}$ represents localization error
 $Error_{class}$ represents class error
and $Error_{confidence}$ represents confidence error

2.2 Backbone network structure

In the process (B) of YOLOv3 algorithm, the backbone network DarkNet-53 can maintain a balance between accuracy and speed. As shown in Figure 3, the basic unit of DarkNet-53 network structure is a convolution layer and residual block, performing residual between different layers of output. The corresponding location of the network is designed with five residual blocks. The residual block structure is shown in Figure 4, showing 1×1 and 3×3 convolutions are used alternately. When the YOLOv3 algorithm runs the DarkNet-53 network, it stops at the last residual block and discards the rest. The DarkNet-53 network has a total of 53 convolutional layers. Due to the removal of the pooling layer, the size conversion of the tensor is downsampling achieved by changing the convolution kernel strides five times.

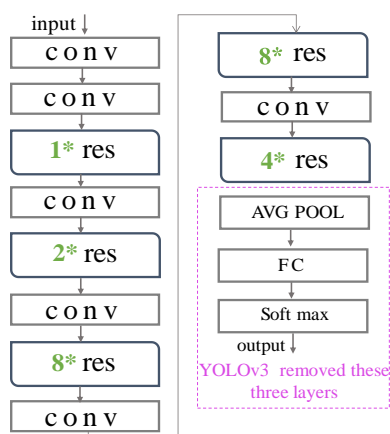


Figure 3. DarkNet-53 Structure

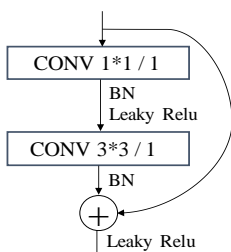


Figure 4. Residual block structure

3. EXPERIMENTS

In this section, the timber-crack dataset construction process is first introduced, including data acquisition, data annotation, and data augmentation; and then the implementation details of timber-crack model training based on the methodology described in Section 2.

3.1 Timber-crack dataset

Due to the special status of architectural heritage, no country has a publicly dataset suitable for timber-crack detection. To overcome this limitation, we have created a real timber-crack dataset.

In terms of data acquisition, an ancient wooden tower in China was selected, which has a history of nearly a thousand years. During the collection process, digital cameras and mobile phones with shooting range of 0.5m-3m were used to obtain a

large number of images distributing in two seasons (i.e., summer and winter) with different light intensity. As the cracking forms of many components are also different, the proportion of various timber-cracks in the dataset needs to be balanced (see Figure 5). According to the above principles, more than 1500 original images were finally collected. The diversity of sample data lays the foundation for the robustness of the model.

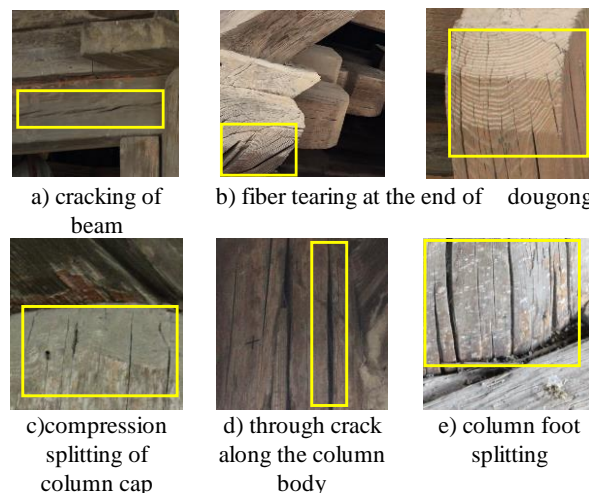


Figure 5. Cracking forms of different components

In terms of data annotation, due to limited GPU memory, high-resolution images need to be first cropped. This experiment was cropped to a size of 800×800 . The cropped images did not overlap and were numbered in order to facilitate subsequent tasks. The *labelimg* software was then used to annotate the location and class of the timber-cracks (see Figure 6).



Figure 6. Data annotation

The annotated image sample data reached 7020, and the data was augmented to 14,040 using mirrored flip method, of which 10960 were set as training set for training models, calculating model gradients and updating weights, 2740 were used as validation set to avoid overfitting and adjust model hyperparameters, 340 were set as test set for reporting the accuracy of the model (Zhang et al, 2019), with a distribution displayed in Figure 7. It shows a balanced distribution of the number of timber-cracks of various types.

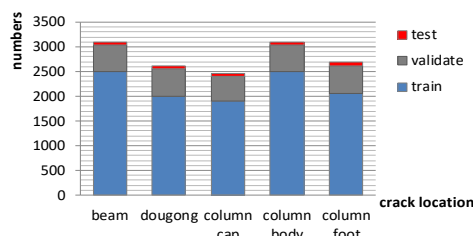


Figure 7. Dataset usage distribution

3.2 Deep learning model learned by YOLOv3

Deep learning technology requires a lot of calculations and requires high computer hardware. This experiment was operated with a configuration of 68G memory and a GPU of 2080TI. The network training and testing work was implemented with Keras 2.1.5 deep learning framework and Python 3.6 language under the Ubuntu 16.04 LTS system. The results were visualized using OpenCV.

The sample size of dataset was 800×800 , and the data was first scaled to 640×640 . In order to facilitate model optimization, training was divided into two stages: pre-training and full-network training. Pre-training refers to training only the last three layers of the network. Full-network training is based on pre-training results to train all layers of the network to obtain the final model. In order to avoid overfitting, the early stopping strategy was adopted to reduce the effective scale of the parameter dimension. In other words, when the performance of the validation set drops, the training is stopped, indicating the preset epoch value is not trained. Table 1 shows the essential initial hyperparameter values while training the model. The optimizer used Adam, which is an effective gradient-based stochastic optimization method, proving to be better than other optimization algorithms in practical applications (Kingma, Ba, 2015).

| Training stage | Hyperparameters | Values |
|-----------------------|-----------------------|--------|
| Pre-Training | Optimizer | Adam |
| | Epoch | 30 |
| | Batch Size | 64 |
| | Initial Learning Rate | 0.001 |
| | Momentum | 0.9 |
| Full-Network Training | Optimizer | Adam |
| | Epoch | 70 |
| | Batch size | 16 |
| | Initial Learning Rate | 0.0001 |
| | Momentum | 0.9 |

Table 1. Initial values of hyperparameters for model training

After the training stage, the model generates three scale feature maps. Since the size of the image entering the network training was set to be 640×640 , the three feature map sizes generated by the convolution operation were: (8×8) , (16×16) , and (32×32) . The small-scale feature map represents a more comprehensive for detecting larger timber-cracks and assigns larger anchorboxes, while the large-scale feature map represents a more detailed image for detecting smaller timber-cracks and assigns smaller anchorboxes (see Table 2). In this study, the anchors used the original size of the YOLOv3 algorithm, which was based on the COCO dataset cluster size.


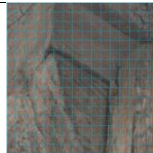

| Category | 8×8 | 16×16 | 32×32 |
|--------------|---|---|---|
| Feature Maps |  |  |  |
| Anchor Size | (116,90) (156,198) (373,326) | (30,61) (62,45) (59,119) | (10,13) (16,30) (33,23) |

Table 2. Multi-scale feature maps generated by the model

4. RESULTS ANALYSIS

The network parameters of the training set are used in the test set, and the qualitative detection results of the test set are shown in Figure 8. It can be seen that the YOLOv3 algorithm can accurately detect timber-cracks of different sizes, and can effectively resist inferior conditions such as different lighting backgrounds in the image. It has good adaptability in the timber-crack detection task.

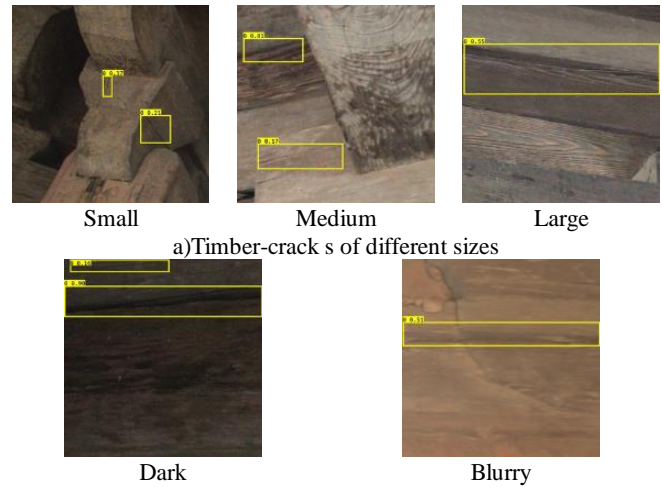


Figure 8. Qualitative results of model testing

In order to quantitatively evaluate the results, the commonly used evaluation indicators were used to report the performance of the model to detect timber-cracks, including speed (S), average precision (AP), precision (P), recall (R), and F_1 -measure (see Table 3). S refers to the time it takes the model to process each image, AP represents the average precision of the model to detect timber-cracks, P and R represent the precision and recall of the timber-crack detection results, and F_1 -measure is the result of comprehensively weighing P and R.

| Indicators | Value |
|----------------|----------|
| S(sec/img) | 0.059272 |
| AP | 0.85317 |
| P | 0.932 |
| R | 0.8784 |
| F_1 -measure | 0.9044 |

Table 3. Quantitative results of model testing

In addition, the P-R curve was used to indicate the correlation between precision and recall (see Figure 9).

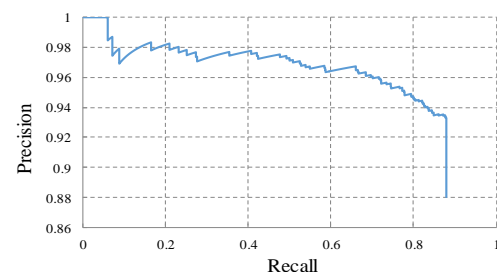


Figure 9. P-R curve

Figure 9 shows the precision of the model trained by the timber-crack dataset and the YOLOv3 algorithm can reach an accuracy of more than 90% with a quick processing speed of less than 0.1s for each image. This promising result assists in fulfilling the tasks of rapidly and accurately detecting the cracks in actual ancient wooden buildings.

5. CONCLUSIONS

This article is aimed at detecting the large number of cracks in wooden architectural heritage in an automatic way. Based on the deep learning technology, two main tasks have been completed. firstly, we collecting crack images in wooden architectural heritage on site, manually labelling them, and constructing a real large-scale timber-crack dataset using data augmentation. Secondly, we training and testing the timber-crack detection model using the YOLOv3 algorithm.

The experimental results show that the timber-crack detection model based on YOLOv3 algorithm has high precision and fast speed, which can be well adapted to the crack detection of wooden architectural heritages. This research illustrates the great potential of applying deep learning technology in the field of heritage protection, providing a new reference scheme for the crack detection of ancient wooden buildings.

ACKNOWLEDGEMENTS

Acknowledgements of support for the Great Wall scholar training program of Beijing Municipal University high level teacher team construction support program (NO. CIT&TCD20180322).

REFERENCES

Dai, J., Chang, L.H., Qian, W., Li, X., 2016: Damage characteristics of ancient architecture wood members and stress wave nondestructive testing of internal void. *Journal of Beijing University of Technology*, 42(02):236-244.

Fu, Y.Z., 2016: Quantitative analysis and repair technology research on the deterioration of wood structure in twentieth century. Beijing University of Technology.

Ji, S.P., Shi, Q.W., Meng, L., 2018: "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set." *IEEE Transactions on Geoscience and Remote Sensing* 57.1: 574-586.

Ju, M.R., Luo, H.B., Wang, Z.B., He, M., Chang, Z., Hui, B., 2019: Improved YOLOv3 algorithm and its application in small target detection. *Acta Optica Sinica*, 39(07):253-260.

Kingma, D., Ba, J., 2015: Adam: A method for stochastic optimization. 3rd International Conference for Learning Representations, 78(1):116-130.

Luo, S.S., 2019: Road target recognition based on Mobile Net-SSD model technical research and its android app development. South China University of Technology.

Redmon, J., et al, 2016: "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition.

Redmon, J., Farhadi, A., 2017: "YOLO9000: better, faster, stronger." Proceedings of the IEEE conference on computer vision and pattern recognition.

Redmon, J., Farhadi, A., 2018: "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767.

Wen, Q., 2019: Research and implementation of building surface crack detection technology based on deep learning. Beijing University of Posts and Telecommunications.

Yang, M., 2016: Study the suitability of the nondestructive detection methods of ancient wood internal defects. Beijing University of Technology.

Zhang, C., et al, 2019: "Raven: A dataset for relational and analogical visual reasoning." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.