Real-Time Inspection System for Ballast Railway Fasteners Based on Point Cloud Deep Learning

HAO CUI[1], **JIAN LI**[2], **QINGWU HU**[3], **AND QINGZHOU MAO**[3]

[1]School of Water Science and Engineering, Zhengzhou University, Zhengzhou 450001, China
[2]School of Geoscience and Technology, Zhengzhou University, Zhengzhou 450001, China
[3]School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China

Corresponding author: Jian Li (lijian5277@163.com)

ABSTRACT Rail fasteners are the most numerous components in railways and they should be inspected periodically. Manual inspection is currently a common solution, which is laborious and low-efficient. Some automatic inspection approaches are proposed. But for ballast railway fasteners inspection, debris, especially ballast along tracks may cover the fasteners, which is still a tricky problem. In this paper, a real-time inspection system for ballast railway fasteners based on point cloud deep learning is developed. Dense and precise point cloud of fastener is obtained from the structured light sensors in the system. The point cloud of fastener is segmented into different parts to avoid the interference of debris on fasteners. A ballast fastener point cloud semantic segmentation dataset is created based on automatic annotation method. Several deep learning point cloud segmentation models are tested in this dataset and PointNet++ is selected to be deployed in the real-time deep learning module of the system. Field tests on ballast railways show excellent accuracy and efficiency of this system.

INDEX TERMS Ballast railway fasteners, fastener inspection, deep learning, neural network, point cloud semantic segmentation.

I. INTRODUCTION

Railway is an important infrastructure for transportation, comparing with other transportation modes, railway transportation achieves a perfect balance in price, efficiency and capacity. Rail fasteners are the most numerous parts in a railway line. In general, there are 4 fasteners in a sleeper and the distance between consecutive sleepers is about 0.6 m, so, there are about 6700 fasteners in 1km railway line. The main functions of rail fasteners are fastening rail tracks onto sleepers or track slab [1], [2], maintain track gauge [3], [4] and reduce vibration (ballastless highspeed railway) [5]–[7]. Rail fastener failures, such as missing or broken will greatly threaten safe operation of trains [8]. Therefore, they must be inspected periodically. In traditional way, rail fasteners are inspected manually, which is laborious and time-consuming. With the increase of labor cost and expansion of railway lines, there is an urgent need for automatic rail fastener inspection method.

The associate editor coordinating the review of this manuscript and approving it for publication was Kathiravan Srinivasan .

In recent years, many researchers have developed various automatic rail fastener inspection methods. Most methods are based on two-dimensional vision like images or video, Marino *et al.* [9] built a real-time visual inspection system for hexagonal-head fasteners based on line-scan camera, this system can detect presence/absence of fasteners with an accuracy of 99.6% in detecting visible fasteners and of 95% in detecting missing fasteners. De Ruvo *et al.* [10] presented a fully automatic and configurable real-time vision system able to detect the presence/absence of fasteners at 187 km/h with the help of Graphic Processing Unit (GPU) parallel processing. Xia *et al.* [11] proposed an efficient method to detect and recognize broken fastener with railway images based on AdaBoost-based algorithm. Yang *et al.* [12] presented a direction field-based method based on images, this method can be used to detect the absence of fasteners at 400 km/h. Resendiz *et al.* [13] proposed a machine vison system consisting of field-acquired video and subsequent analysis, which can automatically detect defects in fasteners and turnout components. Feng *et al.* [14] proposed an automatic visual inspection system for detecting partially worn and completely

missing fasteners using probabilistic topic model, which can simultaneously model diverse types of fasteners with different orientations and illumination conditions using unlabeled data. Gibert *et al.* [15] proposed an image-based fastener inspection system using the histogram of oriented gradients features and a combination of linear Support Vector Machine (SVM) classifiers, this system can inspect missing or defective fastener problems with a probability of detection of 98% and a false alarm rate of 1.23%. The methods above are all based on two-dimensional vision. To get whole view of a fastener, these methods acquire vision data from the top, making them unable to detect loose fasteners. In addition, two-dimensional vision is susceptible to interference from rust and stains on fastener surface.

Inspired by vibration-based structural damage inspection method, some scholars have carried out fastener inspection method based on vibration. Valikhani and Younesian [16] proposed a fastener inspection method based on vibration, an accelerometer is installed on the rail track and its vibration signal is aroused by hitting a hammer to the track, the looseness of fasteners is inspected by conditional entropy criterion and the Morlet wavelet transformation. Wei *et al.* [17] developed a vibration-based fastener inspection system, the acceleration signals are analyzed by wavelet packet analysis and the location and severity of loose fastener can be worked out. Vibration-based fastener inspection method need pre-install accelerometer on rail track and arouse vibration signal by hammer, which is very low efficiency and not practical.

Some researchers developed fastener inspection method based on three-dimensional (3D) point cloud from structured light sensors. These sensors can acquire 3D point cloud without interference from rust and stains on fastener surface. Comparing with 2D vision data, 3D point cloud can achieve more fastener inspection applications, such as loose fastener detection [18] and fasteners geometric parameters measurement [19]. Zhang *et al.* [20] developed a structured-light-based fastener inspection system, with a recognition method based on neural network, the system achieved good performance in inspection missing fasteners. Lorente *et al.* [21] presented a fastener inspection system using structured light sensors to evaluate rail gauge and detect missing fasteners with iterative closest point (ICP) algorithm. Aytekin *et al.* [22] developed a real-time rail fastener detection system to detect missing hexagonal-headed fasteners with structured light sensors, in this system, an extensive analysis of various methods based on pixelwise and histogram similarities was conducted. In our previous work [18], [19], a structured-light-based fastener inspection system called Intelligent Rail Checker (IRC) was developed, this system could not only detect missing, defected and loose fasteners, but also measure geometric parameters of fasteners. This system can achieve 99.8% accuracy in detecting missing and defected ballastless railway fastener. These systems above can achieve good accuracy in ballastless railway fasteners, but their accuracy in ballast rail fasteners is lower, main reason causing this problem is that there are a lot of debris, especially ballast on the fasteners
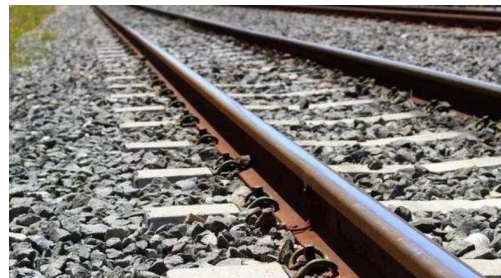

FIGURE 1. Ballast railway line.

in ballast railway, as it shows in FIGURE 1. Ballast railway means the subgrade of rail track is ballast, usually used in railway below 250km/h. For high-speed railway whose speed is higher than that, the subgrade of rail track is replaced by track board. So, high-speed railway can be called ballastless railway.

Ever since Krizhevsky *et al.* [23] proposed AlexNet, which focus on image classification with deep convolutional neural networks, deep learning is flourishing in recent years. From the outset the Deep Learning (DL) models proposed were focusing on image classification or segmentation, such as VGGNet [24], GoogLeNet [25], Fully Convolutional Networks (FCN) [26]. Subsequently, DL models focusing on point cloud classification and segmentation, such as VoxelNet [27], PointNet [28], PointNet++ [29], PointCNN [30], were developed. Many researchers have successfully applied deep learning in many practical applications. Zhang *et al.* [31] used improved GoogLeNet and Cifar10 models to identify maize leaf diseases. Pan *et al.* [32] developed an unmanned surveillance system to observe water levels based on convolutional neural network (CNN). Haggag *et al.* [33] proposed a hybrid DL algorithm for tomato-sorting controllers, this algorithm utilize a combination of CNN and artificial neural network (ANN).

Many researchers had applied DL technic in railways. Faghih-Roohi *et al.* [34] proposed a Deep Convolutional Neural Network (DCNN) solution to the analysis of image data for the detection of rail surface defects. Gibert *et al.* [35] developed an approach to visual track inspection using material classification and semantic segmentation with DCNN, which allows for the detection of crumbling and chipped tie conditions. Yin and Zhao [36] proposed an automated diagnosis network of vehicle on-board equipments for high-speed train via a deep learning approach. Liu *et al.* [37] put forward an automatic fault detection system for the loosed strands of the railway isoelectric line using Faster region-based convolutional neural network and Markov random field model. Wei *et al.* [38] proposed a fastener defect detection method based on image processing and deep learning networks.

Since DL have been applied successfully in many classification and segmentation tasks, we decided to use DL-based method to achieve higher accuracy in ballast railway fasteners inspection. In ballast railway, debris, especially ballast

FIGURE 2. The overview of IRC system.

FIGURE 3. Detailed architecture of the system.

FIGURE 4. Original point cloud of the system.

may partially cover fasteners. It will be quite difficult to classify fasteners with debris and broken fasteners directly. Therefore, we decide to segment the point cloud of fasteners into different parts to avoid the interference of debris on fasteners. We modify our previous IRC system to conducted DL-based point cloud segmentation and ballast fastener inspection in real-time. An automatically fasteners' point cloud annotation method is proposed to offer dataset for DL-based point cloud segmentation models, data augmentation is also used to expand the dataset. Several DL-based point cloud segmentation models are evaluated in this dataset and we chose the one with best accuracy running in modified IRC system. Finally, field tests are conducted on ballast railway lines to verify the accuracy and efficiency of our system.

II. SYSTEM ARCHITECTURE

A. SYSTEM OVERVIEW

The overview of IRC system is shown in FIGURE 2. The yellow cuboid box is the main body of the system, where most sensors are installed. Two white and red cuboid boxes on the yellow box are batteries for the system. IRC system is equipped with quick disassembly structures and pushed forward on rail tracks manually. The quick disassembly structures have four bearing wheels and four limit wheels on inner side of the rail tracks. The two limit wheels on the right side equipped with pneumatic telescopic rods to ensure all the limit wheels are closed to rail tracks, preventing the yellow box from lateral displacement when gauge changes. The system is suitable for 1435mm standard gauge and the pneumatic telescopic rods make the system adapted to ±20mm gauge change. The whole system weighs about 40kg (including two batteries) and can be assembled and disassembled by two workers in 1 minute. An encoder is installed in left bearing wheels to trigger the structured light sensors installed in the yellow box, generating dense point cloud of fasteners when pushing forward.

B. DETAILED ARCHITECTURE

Detailed architecture of the fastener inspection system is shown in FIGURE 3. There are four commercial structured light sensors in the system, corresponding to four fasteners in one sleeper. Two consecutive structured light sensors are
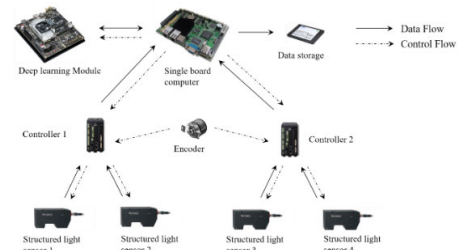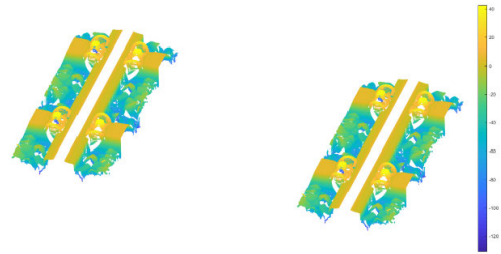
controlled by a controller which is triggered by the encoder installed in left bearing wheels. All the controllers are controlled by a single board computer where the data from the structured light sensors can be processed. The processed data will be sent to deep learning module where point cloud of fasteners is segmented and will also be sent to data storage module. Data and control transmission in this system are all through Ethernet. The controllers are triggered every 1 mm, generating dense point cloud of fasteners, show as FIGURE 4 (the unit is millimeter).

C. MODULES OF THE SYSTEM

In this part, we provide details of the modules in the system, including structured light sensors, encoder, single board computer and deep learning module.

1) STRUCTURED LIGHT SENSORS

The structured light sensors used in this system is a commercial structured light sensor from Keyence, more specifically, Keyence LJ-V7000. The maximum triggering frequency of this sensor is 4000 Hz. The scan range of the sensor is a sector of 30 degrees below, the minimum scan distance is 155mm and the maximum scan distance is 445mm. The repeatability in the vertical and horizontal directions are 0.005mm and 0.06mm, respectively. The distance between consecutive points in a frame is 0.3mm. Generally speaking, the farther scan range of structured light sensor, the lower the accuracy. This structured light sensor gets a perfect balance between scan range and accuracy in terms of rail fastener scanning.

2) ENCODER

The encoder is installed in the left bearing wheels and it will generate triggering signal to two controllers of structured

light sensors. The trigger distance of the encoder is 1mm, it means the distance between consecutive data frame is 1mm, regardless of the movement speed of the whole system. The distance between consecutive frames is 1mm and the distance between consecutive points in a frame is 0.3mm. So, the point cloud of fasteners is very dense (about 330 points per cm^2) and precise, which offers a good foundation for following process.

3) SINGLE BOARD COMPUTER
The single board computer (Intel ATOM D2550@1.86 GHz) is the brain of the whole system. It is running Window XP operation system and online processing program is implemented inside. It sends control signal to the controllers of structured light sensors and receives data from the controllers. The received data will be stored into the data storage module and can also be processed in the single board computer. The point cloud of fasteners is extracted from the original point cloud and sent to the deep learning module for segmentation. The single board computer is also connected to a display panel to interact with users.

4) DEEP LEARNING MODULE
The deep learning module used in this system is Jetson TX2, which is an embedded AI computing device from NVIDIA. It is NVIDIA Pascal architecture with 256 cuda cores inside and 8 GB graphics memory. It has 2 Denver 64-bit CPUs and Quad-Core A57 Complex. The module is running Ubuntu 16.04 operation system, CUDA version up to 9.0, and cuDNN version up to 7.0. Jetson TX2 is a small size, power-efficient, and powerful AI computing device where most deep learning models can run inside. The point cloud of fasteners will be segmented in this module.

III. BALLAST RAILWAY FASTENERS INSPECTION METHODOLOGY
The overall steps of ballast railway fasteners inspection method are shown in FIGURE 5. Firstly, a point cloud extraction algorithm is proposed to extract point cloud of fastener from the original point cloud. Then a fastener point cloud automatic annotation method based on region growing is proposed to create a ballast rail fastener semantic segmentation dataset. After that, three deep learning point cloud segmentation models are tested on this dataset and the one achieves best accuracy is deployed into deep learning module. At last, fastener inspection method after point cloud segmentation is presented to get final results of fastener inspection.

A. FASTENER POINT CLOUD EXTRACTION
In the original point cloud of IRC system, the point cloud of ballast is also captured. To exclude the interference of unrelated point and reduce computational cost, the point cloud of fasteners should be extracted. The 3-dimensional coordinates of fasteners' point cloud are established by 2-dimensional point cloud from the structured light sensors and point cloud
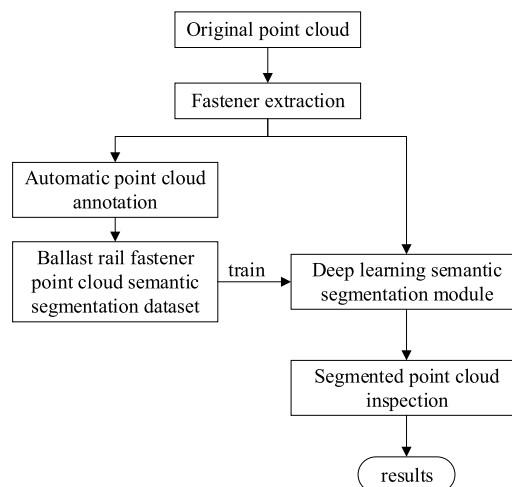

FIGURE 5. The overall steps of ballast rail fasteners inspection methodology.
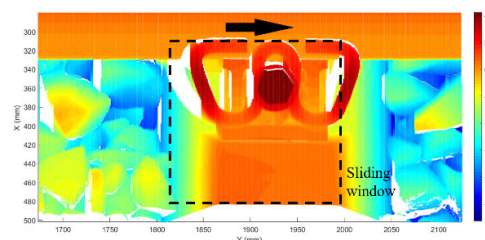

FIGURE 6. Fastener point cloud extraction based on sliding window.

frame number. So, the fasteners' point cloud can be seen as matrix structure, where frame number is matrix row, point number in a frame is matrix column, and height of point is the value of the matrix at the corresponding position. In this matrix structure, the point cloud of fasteners can be located fast and precise.

Rail fasteners are located next to the bottom of rail track and they are higher than surrounding things like rail pad or ballast. The distance between consecutive fasteners is 0.6m, which is a fixed number. The relative distance between fasteners and track bottom in the original point cloud is relatively fixed. Considering above characteristics of rail fasteners, we proposed a sliding-window-based fastener point cloud extraction method, which is shown in FIGURE 6.

The types of most widely used ballast fastener in China is WJ5 and WJ2. The width of sliding window is equal to the width of these two types, which is 150mm. The length of sliding window is set to 160mm. The location of sliding window in X direction is shown in FIGURE 6, the sliding window slides along Y direction. Then calculate the average height of valid point in the sliding window.

To set the height threshold of sliding window, the original output of WJ5 and WJ2 fasteners is selected to calculate average height in sliding window. The result is shown in FIGURE 7, FIGURE 7 (a) and FIGURE 7 (b) are result of WJ5 and WJ2 respectively. In this figure, vertical axis is
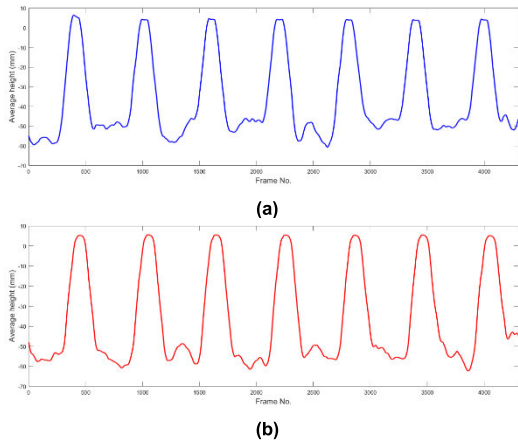
FIGURE 7. Average height in sliding window, (a) Type WJ-5 fastener, (b) Type WJ-2 fastener.
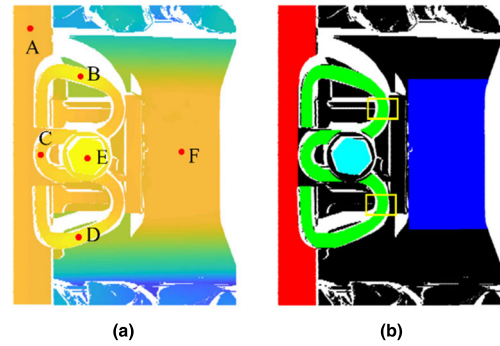

(a)   (b)

FIGURE 8. Fastener point cloud annotation based on region growing (type WJ5), (a) original point cloud and the location of growing seeds, (b) annotation result.

average height and horizontal axis is the frame number where the center of sliding window located. The average height in sliding window cyclical increase when there is a fastener inside. The curve of average height between peaks fluctuates greatly, causing by messy ballast. The height threshold of sliding window is set to -15mm, the frame numbers of adjacent intersections between the average height curve and the threshold are picked out, the mean value of frame numbers is the center frame of fastener. 150 frames before and after the center frame are selected as the point cloud of a fastener.

B. POINT CLOUD SEMANTIC SEGMENTATION DATASET OF BALLAST RAILWAY FASTENERS

Training of deep learning neural networks for point cloud segmentation requires massive labeled data. Therefore, ballast railway fasteners point cloud semantic segmentation dataset should be established first. The original data of the dataset comes from the extracted point cloud of fasteners above, where the distance between adjacent frames is 1mm and the distance between adjacent points in one frame is 0.3mm. The point cloud is distributed unevenly. To remove redundant point and reduce the amount of data, the point cloud in same frame is sampled one point in every three points, generating 1mm*0.9mm point cloud grid.

The next step is data annotation, which is labor consuming and mostly done manually. To avoid manual data annotation, a region growing based point cloud annotation method is proposed. The location of growing seed and the limitation of growing is key point of region growing method. FIGURE 8 (a) shows the location of growing seeds. Seed A is located in the rail bottom to annotate the point cloud of rail bottom. Seed *B*, *C*, and *D* is on upper, middle, and lower part of the spring bar. The occlusion of the fastener itself cause discontinuous of the spring bar's point cloud. So, it's necessary to set three seeds on the spring bar. Seed *E* is on the middle bolt of fastener. Seed *F* is on the shoulder of sleeper. The region growing algorithm used is the same as the one

proposed in our previous work [19]. The growing threshold of spring bar is set to 0.9mm and the threshold of other parts is set to 0.5mm. In FIGURE 8 (b), red part is point cloud of rail bottom, green part is point cloud of spring bar, azure part is point cloud of middle bolt, blue part is point cloud of sleeper shoulder, and black part is unlabeled background. For type WJ2 fasteners, the annotation process is the same, only the location of growing seeds is slightly different.

In general, there are three types of fastener defect, skewed spring bar, missing components, and partial fracture spring bar. Skewed spring bar is caused by the looseness of middle bolt, when the middle bolt loose to much or missing, spring bar will be missing as well. The curved part that is in contact with the base of the fastener (yellow boxes in FIGURE 8(b)) is subjected to huge stress, the spring bar is most likely to fractured in these places. These defects will cause the spring bar to lose effect, which will lead to fastener failure eventually.

To simulate defect fasteners in actual scene and add extra data to the dataset, we expand the dataset by rotating and cutting the point cloud of fasteners. Skewed spring bar is simulated by rotating the point cloud of spring bar around the center of middle bolt following (1) and (2). In these two formulas, (x, y) is the original coordinate of point cloud, (rx_0, ry_0) is the center of rotation, β is the rotating angle clockwise, (x_0, y_0) is the coordinate of point cloud after rotation. The point cloud of spring bar is rotated for 15°, 30°, 180° clockwise and 15°, 30° counterclockwise.

$$x_0 = (x - rx_0)cos\,\beta + (y - ry_0)sin\,\beta + rx_0 \quad (1)$$

$$y_0 = (-x + rx_0)sin\,\beta + (y - ry_0)cos\,\beta + ry_0 \quad (2)$$

To simulate partial fracture spring bar fasteners, the point cloud of spring bar is cut in the curved part. To simulate miss components fasteners, the point cloud of spring bar and middle bolt is all removed. The diagram of extra samples shows in FIGURE 9.

The whole dataset has 1000 normal fasteners, the number of WJ5 fastener is the same as WJ2 fasteners, 4500 simulated defect fasteners (5 rotation situations, 2 cutting situations and 2 removal situations for 250 random selected normal
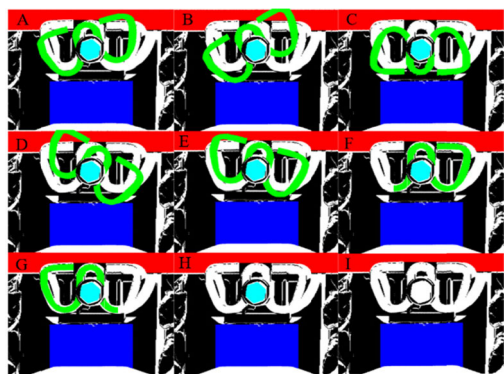
FIGURE 9. Samples of defect fastener simulation (type WJ5).

fasteners of each kind). The whole dataset has 5500 fasteners, 20% of fasteners in each case are random selected as testing set, and the rest fasteners are training set. There are about 60000 points in the point cloud of one fastener, they are random sorted and save as text file, attributes of each point list as x, y, z coordinate and label. The length, width and height range of the point cloud of one fastener is 0 to 0.3m, 0 to 0.24m, and −0.1m to 0.25m respectively.

C. FASTENER POINT CLOUD SEMANTIC SEGMENTATION BASED ON DEEP LEARNING

Deep learning point cloud semantic segmentation methods have achieved far better performance than traditional methods like clustering, RANSAC, region grow, and Hough transform. There are three most used point cloud segmentation benchmarks, S3DIS [39], Semantic3D [40], and ScanNet [41]. S3DIS is a dataset benchmark including point cloud, mesh, RGB image, and surface normal of large-scale indoor scene. Semantic3D has more than 4 billion points of outdoor scene, aiming at point cloud classification and outdoor semantic segmentation. ScanNet is captured by RGB-D camera, has more than 1500 indoor scene point cloud with color, aiming at indoor scene semantic segmentation. It is the most widely used indoor point cloud segmentation benchmark, but most deep learning models haven't used the color in point cloud. The ballast railway fastener point cloud semantic segmentation dataset has unordered point cloud and individual fastener can be seen as an indoor scene, so, it is most similar to ScanNet. Three models that achieve high point cloud semantic segmentation accuracy on ScanNet are chosen to evaluate their performance on the ballast fastener point cloud semantic segmentation dataset.

1) POINTNET++

PointNet++ [29] is an extension of PointNet which is novel type of neural network that directly consumes point clouds. PointNet either transforms features of individual points independently or process global features of the entire point set. However, in many cases there are well defined distance metrics such as Euclidean distance for 3D point clouds collected

by 3D sensors or geodesic distance for manifolds like isometric shape surfaces. PointNet++ learns hierarchical features with increasing scales of contexts, just like that in convolutional neural networks. To deal with non-uniform densities of point cloud, special layers that are able to intelligently aggregate information from different scales are proposed. PointNet++ can achieve overall accuracy of 84.5% in semantic segmentation of ScanNet.

2) POINTSIFT

Inspired by the outstanding 2D shape descriptor SIFT, Jiang *et al.* design a module called PointSIFT [42] that encodes information of different orientations and is adaptive to scale of shape. Specifically, an orientation-encoding unit is designed to describe eight crucial orientations, and multi-scale representation is achieved by stacking several orientation-encoding units. PointSIFT module can be integrated into various PointNet-based architecture to improve the representation ability. PointSIFT can achieve overall accuracy of 86.2% in semantic segmentation of ScanNet.

3) POINTCNN

PointCNN [30] is a simple and general framework for feature learning from point cloud. Point clouds are irregular and unordered, thus directly convolving kernels against features associated with the points will result in desertion of shape information and variance to point ordering. To address these problems, an *X*-transformation is proposed to learn from the input points to simultaneously promote two causes: the first is the weighting of the input features associated with the points, and the second is the permutation of the points into a latent and potentially canonical order. Element-wise product and sum operations of the typical convolution operator are subsequently applied on the *X*-transformed features. The proposed method is a generalization of typical CNNs to feature learning from point clouds, thus it is called PointCNN. Experiments show that PointCNN achieves overall accuracy of 85.1% in semantic segmentation of ScanNet.

These neural networks will be evaluated on the ballast railway fastener point cloud semantic segmentation dataset in next section. The one achieves best performance will be implanted in the Deep learning module of IRC. The segmented point cloud of fastener will be inspected to find out defected fastener.

D. FASTENER INSPECTION BASED ON ITS SEGMENTED POINT CLOUD

After point cloud semantic segmentation by deep learning module, the point cloud of fastener will go through several steps to inspect defects.

First step is checking the absence of middle bolt of fastener. Middle bolt is the most important component of a fastener, if middle bolt is missing, the fastener will lose its function. The middle bolt is inspected by checking the number of its points. In the neural networks above, the input point cloud will be sampled into a certain number to get regularized input
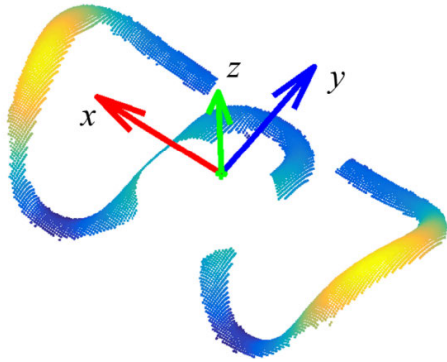
FIGURE 10. Spring bar posture evaluation based on PCA.

TABLE 1. Software environments of deep learning models.

Models	TensorFlow	CUDA	cuDNN
PointNet++	1.4	8.0	5.1
PointSIFT	1.4	8.0	5.1
PointCNN	1.6	9.0	7.0

TABLE 2. The segmentation accuracy of deep learning models in our dataset.

class	PointNet++	PointSIFT	PointCNN
OA	99.29%	99.33%	**99.39%**
Rail bottom	99.82%	99.75%	**99.89%**
Spring bar	**99.37%**	99.31%	99.25%
Middle bolt	**97.98%**	97.91%	97.95%
Sleeper shoulder	96.58%	**97.25%**	97.12%

that adapted to network structure. Therefore, the segmented point cloud of fasteners will be less than input points. Assume that the sampling rate of input points is δ, the number of segmented points has to divided by δ. The area of middle bolt upper surface is different between WJ5 and WJ2. So, the number of its points is different as well, a normal middle bolt of WJ5 fastener has 800 points while that of WJ2 has only 300 points. The threshold of the number of middle bolt point cloud is set to 400 for WJ5 fastener and 150 for WJ2 fastener, the middle bolt will be considered missing if the number of its point cloud is lower than its corresponding threshold.

Spring bar is also an important component for a fastener, it should not only go through point number check, but also posture check. A normal WJ5 and WJ2 fastener has 5500 and 4450 points in its spring bar respectively. The threshold of the number of spring bar point cloud is set to 5000 for WJ5 fastener and 4000 for WJ2 fastener, if the number of spring bar points is smaller than the threshold, the spring bar will be considered partial fractured. Otherwise the spring bar is considered complete, the posture of spring bar will be evaluated by Principal Component Analysis (PCA), as shows by FIGURE 10. Then calculate the deviation between the main direction of spring bar (blue arrow in FIGURE 10) and the y axis of fastener coordinate, if the deviation angle is larger than 5°, the spring bar will be considered skewed.

IV. EXPERIMENTS

Experiments include two parts, the first part is performance evaluation of deep learning models on ballast railway fastener point cloud segmentation dataset, the other part is field tests of the fastener inspection system on ballast railway lines.

A. PERFORMANCE EVALUATION OF DEEP LEARNING MODELS ON OUR DATASET

The performance of deep learning models is evaluated on our ballast rail fastener point cloud segmentation dataset. The experiment platform is a PC with Ubuntu16.04 OS, Intel Xeon W-2145 CPU@3.7GHz, 64GB RAM, and NVIDIA Titan RTX GPU with 22GB RAM. The software environments of deep learning models are shown in TABLE 1. The parameters about size in the models are scaled down to fit

the size of fastener. Training epoch and batch size of these models are set to 500 and 32, respectively. Other parameters like training steps, learning rates, and decay rates are all remain the default settings of each model. The training time for PointNet++, PointSIFT, and PointCNN is 51 hours, 35 hours, and 45 hours, respectively.

The index of accuracy used are class accuracy (CA) and overall accuracy (OA), CA is worked out through (3), which is all the correctly predicted points of this class divided by the all the predicted points of this class. OA is calculated with (4), which is correctly predicted points of all classes divided by all the points in this fastener.

$$CA = \frac{TP}{TP + FP} \tag{3}$$

$$OA = \frac{\sum_{i=1}^{k} TP}{\sum_{i=1}^{k} (TP + FP)} \tag{4}$$

The accuracy of deep learning models on ballast rail fastener segmentation dataset shows in TABLE 2. Since rail fasteners are standard parts, the size and location change of fastener components in the point cloud is rather small, all the deep learning models achieve very high accuracy in our dataset. PointCNN achieves highest OA and CA in rail bottom, PointSIFT gets highest CA in sleeper shoulder, and PointNet++ achieves highest CA in spring bar and middle bolt. Only the point cloud of spring bar and middle bolt is used in the following fastener inspection, so, PointNet++ is selected and deployed into Jetson TX2 deep learning module.

FIGURE 11 shows the segmentation result of PointNet++ on skewed fasteners in actual scene. The segmentation result is basically correct. There are lots of holes in the point cloud since input point cloud is sampled into a fixed number to get regularized input that adapted to network structure. FIGURE 12 shows the segmentation result of PointNet++ on abnormal fasteners in actual scene. FIGURE 12 (a), (c) are ballast shielding fasteners, the ballast is marked out in red
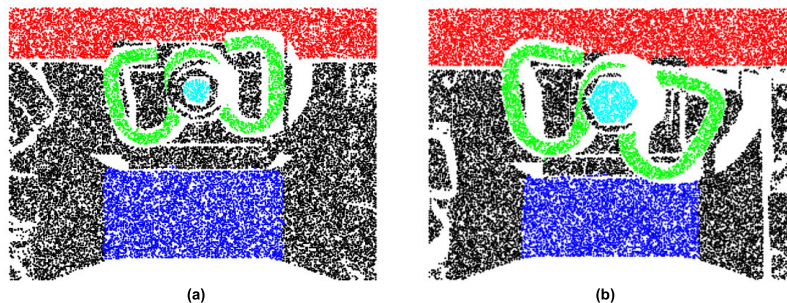
FIGURE 11. Segmentation result of PointNet++ on skewed fasteners in actual scene, (a) skewed WJ2 fastener, (b)skewed WJ5 fastener.
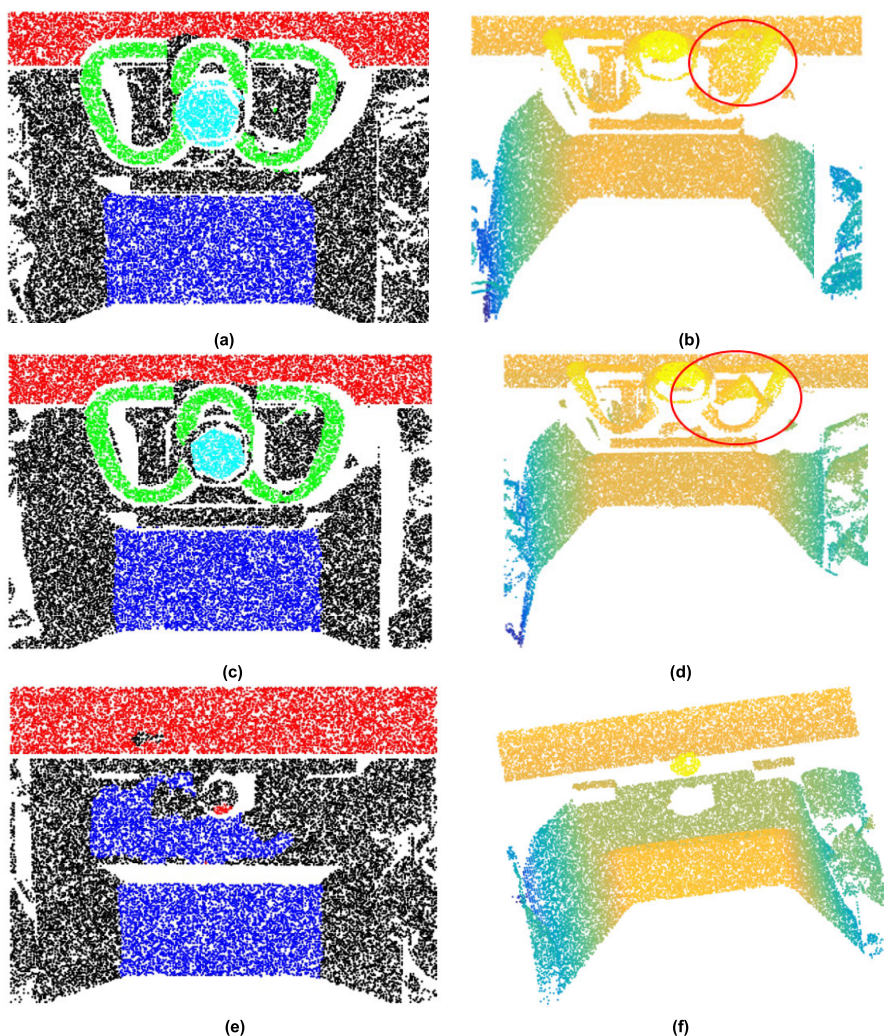

FIGURE 12. Segmentation results of PointNet++ on abnormal fasteners in actual scene, (a) ballast shielding fasteners, (b) side view of a, (c) ballast shielding fasteners, (d) side view of c, (e) spring bar missing fastener, (f) side view of e.

frame in FIGURE 12 (b), (d) respectively. The point cloud of ballast and spring bar is clearly segmented. FIGURE 12 (e) is spring bar missing fastener, sleeper shoulder is over segmented and middle bolt is not labeled, but no spring bar point is labeled, which will still be classified as defected fastener in following inspection process. PointNet++ shows good

generalization ability in new fasteners that not in the ballast railway fastener segmentation dataset.

To test the time efficiency of trained PointNet++ model, it is deployed into Jetson TX2, every parameter is the same as default setup except for the batch size is set to 4, since there are 4 fasteners in a sleeper. The evaluation of whole ballast

TABLE 3. The confusion matrix between IRC inspection result and manual inspection benchmark.

		IRC inspection result (decision tree-based method/method in this paper)			
		Normal	Skewed	Fractured	Missing
Manual Inspection result	Normal	11549/12036	458/10	59/17	3/4
	Skewed		15/15		
	Fractured			11/15	
	Missing				9/7
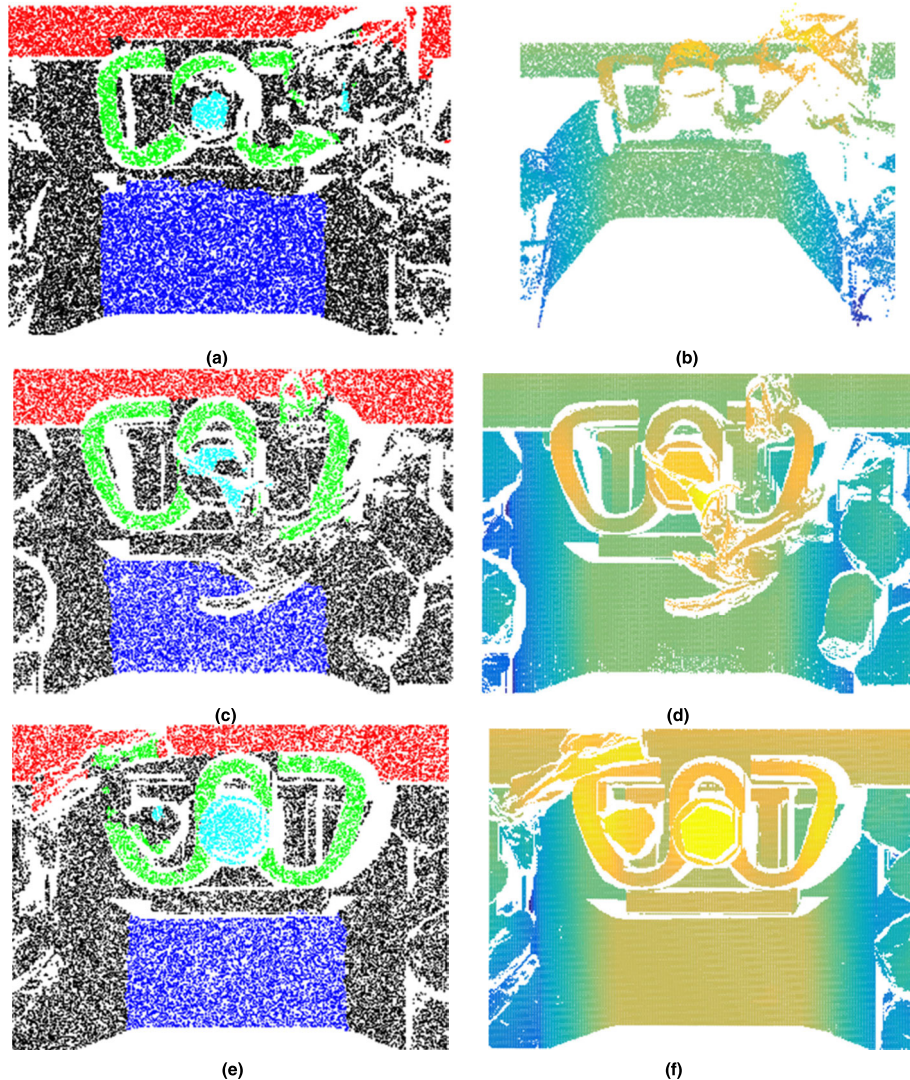
FIGURE 13. Misclassified samples by segmentation-based fastener inspection method, (a) normal WJ2 fastener covered by debris wrongly classified as fractured one, (b) side view of a, (c) normal WJ5 fastener covered by debris wrongly classified as missing component, (d) top view of c, (e) normal WJ5 fastener covered by debris wrongly classified as skewed one, (f) top view of e.

rail fastener segmentation dataset takes 20s, considering the dataset has 5500 fasteners, which is 1375 batches, it takes 15ms for one batch.

B. FIELD TESTS OF THE FASTENER INSPECTION SYSTEM
Before field tests of the fastener inspection system, the time efficiency of the system should be tested first to verify its

real-time processing ability. While the dataset preparation process, we got much original point cloud of fasteners. A 2km railway line was selected and went through fastener inspection process by the system. The whole inspection process, including fastener point cloud extraction, segmentation, and inspection took 305.2 seconds. That is to say the inspection speed of the system is 6.6 m/s, which is much faster than walking speed. This fastener inspection system will ensure real-time processing while pushing manually along rail tracks.

Two field tests of the fastener inspection system were conducted on ballast railway lines in Wuhan, China. These railway lines are new lines and the fasteners are not in the fastener point cloud segmentation dataset. One test was conducted on railway line with WJ5 fasteners, the other test was on railway line with WJ2 fasteners. Both tests were about one kilometer in length. The IRC fastener inspection system was pushed manually on the rail lines during the outage time of train in early morning. The fasteners in test area were inspected manually to get benchmark result and make efficiency comparison with IRC system. Eventually, 5856 WJ2 fasteners and 6248 WJ5 fasteners were inspected by both IRC system and human. The IRC system was pushing in walk speed and real-time classification was conducted on board, total time consumption was 26 minutes. Manual inspection costed 4 workers about 3 hours and a half to complete.

The fasteners were classified into four groups, namely, normal fastener, skewed spring bar fastener, partially fracture spring bar fastener, and missing component fastener. The decision tree based ballastless rail fastener classification method in our previous work [18] was also tested in the same area to make comparison with the segmentation-based ballast railway fastener inspection method proposed in this paper.

The confusion matrix of manual inspection and IRC inspection is shown in TABLE 3. The result of decision tree-based method and segmentation-based method proposed in this paper are both used. Many normal fasteners were classified as skewed and fracture fastener by decision tree-based method, mainly caused by shielding of ballast, as the case in FIGURE 12 (c) and (e). Several normal fasteners were classified as fractured or missing fastener by segmentation-based method proposed in this paper, as is shown in FIGURE 13. The misclassified samples are caused by debris that cover the fastener, FIGURE 13 (a) and (e) is classified as fractured spring bar fasteners, FIGURE 13 (c) is classified as missing component fastener, and FIGURE 13 (e) is classified as skewed spring bar fasteners.

The decision tree based ballastless railway fastener classification method achieves 95.70% accuracy in ballast rail fasteners while segmentation-based method proposed in this paper gets 99.74% accuracy, which is 4.04% higher. In terms of efficiency, the IRC fastener inspection system is more than 30 times faster than manual inspection.

V. CONCLUSION

In this paper, a real-time inspection system for ballast railway fastener based on point cloud deep learning is developed. Dense and precise point cloud of fastener is obtained from the structured light sensors in the system. Since there are lots of debris, especially ballast covering fasteners in ballast railway. Instead of directly classification, the fastener's point cloud is segmented into different parts according to its components to avoid the interference of debris on fasteners. A ballast fastener point cloud semantic segmentation dataset is created via automatic annotation method. Several deep learning point cloud semantic segmentation models is tested in this dataset and PointNet++ is selected to deployed in the deep learning module of the fastener inspection system. This system achieved 99.7% accuracy in field test on ballast railways and its efficiency is far more efficient than manual inspection. In the future, we plan to gather more fastener data to achieve balance between negative and positive samples in the dataset, optimize the deep learning model by applying new activation functions, and apply this model to thinner point cloud for transplanting this system to train platform.

ACKNOWLEDGMENT

REFERENCES

[1] S. Morales-Ivorra, J. I. Real, C. Hernández, and L. Montalbán, "Derailment risk and dynamics of railway vehicles in curved tracks: Analysis of the effect of failed fasteners," *J. Mod. Transp.*, vol. 24, no. 1, pp. 38–47, 2016, doi: 10.1007/s40534-015-0093-z.

[2] L. Zhou and Z.-Y. Shen, "Dynamic analysis of a high-speed train operating on a curved track with failed fasteners," *J. Zhejiang Univ. Sci. A*, vol. 14, no. 6, pp. 447–458, 2013, doi: 10.1631/jzus.a1200321.

[3] T. Tong, Y. Luo, and L. Li, "Research on the influence of rail fastener stiffness upon dynamic performance of vehicle-track system," in *Proc. 7th Int. Conf. Measuring Technol. Mechatronics Automat. (ICMTMA)*, 2015, pp. 923–925, doi: 10.1109/icmtma.2015.227.

[4] C. Esveld, *Modern Railway Track*. Zaltbommel, The Netherlands: MRT-Productions, 2001.

[5] X. Xiao, X. Jin, and Z. Wen, "Effect of disabled fastening systems and ballast on vehicle derailment," *J. Vib. Acoust.*, vol. 129, no. 2, pp. 217–229, 2007, doi: 10.1115/1.2424978.

[6] K. P. Köstli, C. J. C. Jones, and D. J. Thompson, "Experimental and theoretical analysis of railway bridge noise reduction using resilient rail fasteners in Burgdorf, Switzerland," in *Noise and Vibration Mitigation for Rail Transportation Systems*. Berlin, Germany: Springer, 2008, pp. 208–214, doi: 10.1007/978-3-540-74893-9_29.

[7] G.-L. Dai and W.-S. Liu, "Applicability of small resistance fastener on long-span continuous bridges of high-speed railway," *J. Central South Univ.*, vol. 20, no. 5, pp. 1426–1433, 2013, doi: 10.1007/s11771-013-1631-7.

[8] H. Wang, A. Che, S. Feng, and X. Ge, "Full waveform inversion applied in defect investigation for ballastless undertrack structure of high-speed railway," *Tunnelling Underground Space Technol.*, vol. 51, pp. 202–211, 2016, doi: 10.1016/j.tust.2015.10.035.

[9] F. Marino, A. Distante, P. L. Mazzeo, and E. Stella, "A real-time visual inspection system for railway maintenance: Automatic hexagonal-headed bolts detection," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 3, pp. 418–428, May 2007, doi: 10.1109/itsc.2004.1398936.

[10] P. De Ruvo, A. Distante, E. Stella, and F. Marino, "A GPU-based vision system for real time detection of fastening elements in railway inspection," in *Proc. 16th IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 2309–2312, doi: 10.1109/icip.2009.5414438.

[11] Y. Xia, F. Xie, and Z. Jiang, "Broken railway fastener detection based on adaboost algorithm," in *Proc. Int. Conf. Optoelectron. Image Process. (ICOIP)*, vol. 1, 2010, pp. 313–316, doi: 10.1109/icoip.2010.303.

[12] J. Yang, W. Tao, M. Liu, Y. Zhang, H. Zhang, and H. Zhao, "An efficient direction field-based method for the detection of fasteners on high-speed railways," *Sensors*, vol. 11, no. 8, pp. 7364–7381, Jul. 2011, doi: 10.3390/s110807364.

[13] E. Resendiz, J. M. Hart, and N. Ahuja, "Automated visual inspection of railroad tracks," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 751–760, Jun. 2013, doi: 10.1109/tits.2012.2236555.

[14] H. Feng, Z. Jiang, F. Xie, P. Yang, J. Shi, and L. Chen, "Automatic fastener classification and defect detection in vision-based railway inspection systems," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 4, pp. 877–888, Apr. 2014, doi: 10.1109/tim.2013.2283741.

[15] X. Gibert, V. M. Patel, and R. Chellappa, "Robust fastener detection for autonomous visual railway track inspection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2015, pp. 694–701, doi: 10.1109/wacv.2015.98.

[16] M. Valikhani and D. Younesian, "Application of an optimal wavelet transformation for rail-fastening system identification in different preloads," *Measurement*, vol. 82, pp. 161–175, Mar. 2016, doi: 10.1016/j.measurement.2015.12.039.

[17] J. Wei, C. Liu, T. Ren, H. Liu, and W. Zhou, "Online condition monitoring of a rail fastening system on high-speed railways based on wavelet packet analysis," *Sensors*, vol. 17, no. 2, p. 318, 2017, doi: 10.3390/s17020318.

[18] Q. Mao, H. Cui, Q. Hu, and X. Ren, "A rigorous fastener inspection approach for high-speed railway from structured light sensors," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 249–267, Sep. 2017, doi: 10.1016/j.isprsjprs.2017.11.007.

[19] H. Cui, Q. Hu, and Q. Mao, "Real-time geometric parameter measurement of high-speed railway fastener based on point cloud from structured light sensors," *Sensors*, vol. 18, no. 11, p. 3675, 2018, doi: 10.3390/s18113675.

[20] H. Zhang, W. Tao, M. Liu, and H. Zhao, "A laser scanning system for the inspection of fasteners in railways," *Lasers Eng.*, vol. 24, nos. 3–4, pp. 229–240, 2013. [Online]. Available: <Go to ISI>://WOS:000348956100016

[21] A. G. Lorente, D. F. Llorca, M. G. Velasco, J. A. R. Garcia, and F. S. Dominguez, "Detection of range-based rail gage and missing rail fasteners use of high-resolution two- and three-dimensional images," *Transp. Res. Rec.*, vol. 2448, pp. 125–132, 2014, doi: 10.3141/2448-15.

[22] Ç. Aytekin, Y. Rezaeitabar, S. Dogru, and I. Ulusoy, "Railway fastener inspection by real-time machine vision," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 7, pp. 1101–1107, Jul. 2015, doi: 10.1109/tsmc.2014.2388435.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105, doi: 10.1145/3065386.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[25] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9, doi: 10.1109/cvpr.2015.7298594.

[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[27] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3D object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4490–4499, doi: 10.1109/cvpr.2018.00472.

[28] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. IEEE Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, vol. 1, no. 2, p. 4, doi: 10.1109/cvpr.2017.16.

[29] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5105–5114.

[30] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution On X-Transformed Points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 828–838.

[31] L. Zhang, J. Jia, G. Gui, X. Hao, W. Gao, and M. Wang, "Deep learning based improved classification system for designing tomato harvesting robot," *IEEE Access*, vol. 6, pp. 67940–67950, 2018, doi: 10.1109/access.2018.2879324.

[32] J. Pan, Y. Yin, J. Xiong, W. Luo, G. Gui, and H. Sari, "Deep learing-based unmanned surveillance systems for observing water levels," *IEEE Access*, vol. 6, pp. 73561–73571, 2018, doi: 10.1109/access.2018.2883702.

[33] M. Haggag, S. Abdelhay, A. Mecheter, S. Gowid, F. Musharavati, and S. Ghani, "An intelligent hybrid experimental-based deep learning algorithm for tomato-sorting controllers," *IEEE Access*, vol. 7, pp. 106890–106898, 2019 2019, doi: 10.1109/access.2019.2932730.

[34] S. Faghih-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, B. De Schutter, "Deep convolutional neural networks for detection of rail surface defects," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2016, pp. 2584–2589, doi: 10.1109/ijcnn.2016.7727522.

[35] X. Gibert, V. M. Patel, and R. Chellappa, "Material classification and semantic segmentation of railway track images with deep convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 621–625, doi: 10.1109/icip.2015.7350873.

[36] J. Yin and W. Zhao, "Fault diagnosis network design for vehicle on-board equipments of high-speed railway: A deep learning approach," *Eng. Appl. Artif. Intell.*, vol. 56, pp. 250–259, Nov. 2016, doi: 10.1016/j.engappai.2016.10.002.

[37] Z. Liu, L. Wang, C. Li, and Z. Han, "A high-precision loose strands diagnosis approach for isoelectric line in high-speed railway," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 1067–1077, Mar. 2018, doi: 10.1109/tii.2017.2774242.

[38] X. Wei, Z. Yang, Y. Liu, D. Wei, L. Jia, and Y. Li, "Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study," *Eng. Appl. Artif. Intell.*, vol. 80, pp. 66–81, 2019, doi: 10.1016/j.engappai.2019.01.008.

[39] *Stanford 2D-3D-Semantics Dataset*. Accessed: Jul. 15, 2019. [Online]. Available: http://buildingparser.stanford.edu/dataset.html

[40] *Semantic3D*. Accessed: Jul. 11, 2019. [Online]. Available: http://www.semantic3d.net/

[41] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5828–5839, doi: 10.1109/cvpr.2017.261.

[42] M. Jiang, Y. Wu, T. Zhao, Z. Zhao, and C. Lu, "PointSift: A sift-like network module for 3D point cloud semantic segmentation," 2018, *arXiv:1807.00652*. [Online]. Available: https://arxiv.org/abs/1807.00652

• • •