Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network

Chen, Junwen; Liu, Zhigang; Wang, Hongrui; Nunez, Alfredo; Han, Zhiwei

DOI
10.1109/TIM.2017.2775345

Publication date
2018

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Instrumentation and Measurement

Citation (APA)
Chen, J., Liu, Z., Wang, H., Nunez, A., & Han, Z. (2018). Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network. *IEEE Transactions on Instrumentation and Measurement*, *67*(2), 257-269. https://doi.org/10.1109/TIM.2017.2775345

Automatic Defect Detection of Fasteners on the Catenary Support Device Using Deep Convolutional Neural Network

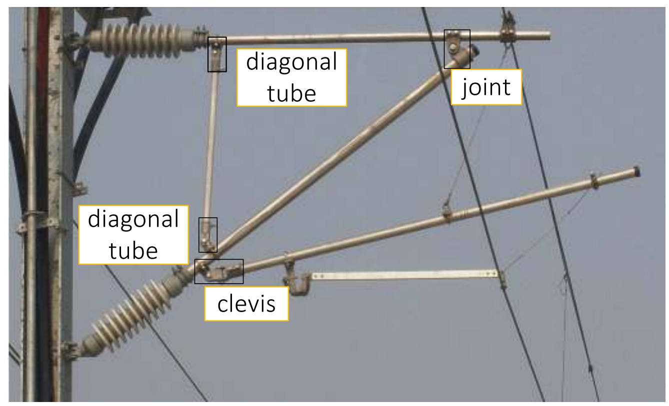
Junwen Chen, *Student Member, IEEE*, Zhigang Liu, *Senior Member, IEEE*, Hongrui Wang, *Student Member, IEEE*, Alfredo Núñez, *Senior Member, IEEE*, and Zhiwei Han, *Member, IEEE*

***Abstract*[1]— The excitation and vibration triggered by the long-term operation of railway vehicles inevitably result in defective states of catenary support devices. With the massive construction of high-speed electrified railways, automatic defect detection of diverse and plentiful fasteners on the catenary support device is of great significance for operation safety and cost reduction. Nowadays, the catenary support devices are periodically captured by the cameras mounted on the inspection vehicles during the night, but the inspection still mostly relies on human visual interpretation. To reduce the human involvement, this paper proposes a novel vision-based method that applies the deep convolutional neural networks (DCNNs) in the defect detection of the fasteners. Our system cascades three DCNN-based detection stages in a coarse-to-fine manner, including two detectors to sequentially localize the cantilever joints and their fasteners and a classifier to diagnose the fasteners' defects. Extensive experiments and comparisons of the defect detection of catenary support devices along the Wuhan-Guangzhou high-speed railway line indicate that the system can achieve a high detection rate with good adaptation and robustness in complex environments.**

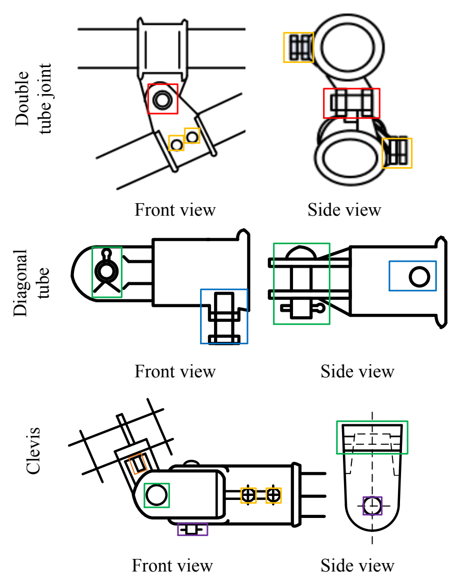
***Index Terms*— High-speed railway, catenary support device, fastener, automatic defect detection, deep convolutional neural network**

I. Introduction

In the electrified railway industry, the pantograph-catenary system plays an important role in transmitting power from the traction network to vehicles. Catenary support device (see Fig. 1) is utilized to maintain the height and stagger of the conductor line, namely the contact wire. However, sophisticated mechanical and electrical interactions exist between the pantograph and catenary, which inevitably cause a high defect rate of the pantograph-catenary system and strongly influence operation safety [1]. Particularly, due to the vibration and excitation in long-term operation, fasteners serving as the connection of the cantilevers on the catenary support devices may loosen, break or are even missing.



(a)



(b)

Fig. 1 Structure of the catenary support device. (a) Structure overview. (b) Installation structure of the cantilever joints. Red, yellow, green, blue, orange and purple boxes indicate the positions of the nut, screw B, α pin, puller bolt, screw A and β pin.

[1] This work was supported by the National Natural Science Foundation of China (U1434203, 51377136, 51407147), as well as the Sichuan Province Youth Science and Technology Innovation Team (2016TD0012). (*Corresponding author: Zhigang Liu*)

J. Chen, Z. Liu, and Z. Han are with the School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China (e-mail: junwenchen@yeah.net; liuzg_cd@126.com; zw.han@my.swjtu.edu.cn;).

H. Wang is with the School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China and the Section of Railway Engineering, Delft University of Technology, Delft 2628CN, The Netherlands (e-mail: soul_wang0@163.com).

A. Núñez is with the Section of Railway Engineering, Delft University of Technology, Delft 2628CN, The Netherlands (e-mail: a.a.nunezvicencio@tudelft.nl).

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

As shown in Fig. 1(a), on the catenary support devices, the four joints (i.e., the double tube joint, clevis and two diagonal tubes) are installed to concatenate the horizontal cantilever, the oblique cantilever, the cantilever arm and the registration arm. According to the China Railway Standard [2], the cantilever joints are fixed by the six different fasteners (i.e., two screws, puller bolt, α-pin, β-pin, and nut), as shown in Fig. 1 (b).

Non-contact detection is widely adopted with the great advances in imaging technology [3]. The railway personnel manually detect the defects by reading a large volume of data from captured images offline. Due to the installation structure, in the shooting angle, defects including the missing and the latent missing of screw A, the puller bolt and α-pin, β-pin, and missing of the big nuts and the top-view screws can be detected.

However, with the massive construction of high-speed railways, the total mileage of China's electrified railway is over 74,000 kilometers. More than 1.03 billion catenary support components must be manually detected. Personnel can easily get vision fatigue and correspondingly miss some defects. Manual detection is performed infrequently, so defects may not be detected in time. Therefore, it is necessary to develop an automatic defect recognition method based on the catenary support device images.

For the power supply system pantograph-catenary, some intelligent detection experiments have been accomplished, such as catenary geometry parameter measurement [4], surface wear diagnosis of the pantograph and contact wire [5] and insulator defects diagnosis [6], by image processing and machine learning. To realize the automatic defect detection of fasteners on the catenary support devices, this paper refers to the pioneering works on railroad track detection. For surface defects of rail heads, Li *et al.* [7] designed a visual detection system to capture the rail road images and extract the discrete defects based on a projection profile. In addition, researchers have proposed some methods for detecting railroad fasteners. Feng *et al.* [8] developed an automatic defect detection method using a probabilistic topic model. Marino *et al.* [9] used a multilayer perception neural classifier to detect missing hexagonal bolts. Aytekin achieved real-time railway fastener inspection using a high-speed laser range finder camera and pixel and histogram similarity analysis [10]. As Deep Convolutional Neural Network (DCNN) [11] prevails in object recognition, Gibert *et al.* [12] applied DCNN in railroad track detection. This multi-task learning system combined a 10-class track material classification detector (e.g., wood, concrete, and metal fasteners *etc.*) with a support vector machine (SVM)-based detector for fastener defects via a fully convolutional neural network and achieved a state-of-the-art result compared to shallow learning. Big data technologies include not only the image processing but also time delay prediction [13][14] and condition based maintenance [15], which make the machine learning technologies promising in the railway system.

Automatic defect detection of fasteners on the catenary support device has not been achieved, to the best knowledge of the authors. The railway track fasteners are usually orderly arranged and firmly fixed on the rail. However, the railway catenary support devices are not uniform. The cantilevers are connected to the masts by hinges, which rotate the support devices into multiple shapes and angles. Due to the large scale and complexity of captured images, the segmentation method of fasteners via the rail material classification [16] cannot be used in the case of catenary support devices. Accordingly, a new DCNN based model is proposed to identify the components in the captured HD images and then judge their states. The system is based on the following pioneer work.

1) Object detection

Recently, various object detection algorithms based on DCNN have become ubiquitous and achieved good results in the vision benchmark [17]. Based on region proposal, Girshick proposed a region convolutional neural network (R-CNN) [18] and Fast R-CNN [19]. Faster R-CNN unifies the region proposal generation and the object classification network into an end-to-end framework [20]. Based on regression, Redmond *et al.* [21] developed a fast single shot detection method named you only look once (YOLO). In the Pascal VOC dataset [17], YOLO can process 45 frames per second without sacrificing accuracy. Liu *et al.* [22] designed a single-shot multi-box detector (SSD) that produces the default boxes for object detection, which offers a speed up compared to the region proposal generation in Faster R-CNN. The DCNN architectures adopt feature learning instead of the traditional hand-crafted feature extraction [23] to improve robustness.

2) Object classification

For image classification, Krizhevsky *et al.* [24] designed AlexNet to classify 1.2 million ImageNet ILSVRC images that belong to 1000 classes. Szegedy *et al.* [25] developed a 22-layer deep network named GoogleNet that achieved state-of-the-art results in 2014. Training strategies such as dropout and weight decay play important roles in preventing overfitting.

3) Cascaded DCNN

Cascaded DCNN has been proposed in scene text segmentation [26], face detection and finger detection [27], *etc.* to improve efficiency in a greedy manner. Particularly in face detection and the alignment field, cascaded DCNN is widely used. In reference [28], the DCNNs are cascaded to detect the facial points in the input face images. Zhang *et al.* [29] built a three-stage DCNN to detect the faces and facial points successively.

It can be seen in Fig. 1 that the number and class of the fasteners are fixed in the cantilever joints. In analogy to face detection, the cascade structure can be adopted in our task to detect the cantilever joints and the fasteners, and classify the states of fasteners from coarse to fine.

This paper is organized as follows. The overview of the defect detection system is given in Section II. The cascaded DCNNs are theoretically described and selected for the localization of the cantilever joints and their fasteners and the recognition of the defective fasteners in Section III. Section IV presents the adopted dataset of catenary support device images and analyzes the advantages of the detection method by several

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

experiments and comparisons. Section V draws some conclusions and outlines further improvements.

II. System Overview

The catenary support device is captured by the roof-mounted cameras on the running vehicle (see Fig. 2). To avoid the interference of background buildings, the images are obtained during night work. The cameras continuously photograph the catenary support devices in global and local views from both the front and reverse sides. The size of the catenary support device images is 6600×4400 pixels. The location information such as the number and mileage mark of the captured catenary support device are recorded in the vehicle database. The image processing consists of three major stages in a coarse-to-fine manner, component extraction, fastener extraction and the fastener state classification. Fig. 3 describes the pipeline of the detection module. Overviews of the three stages are as follows.

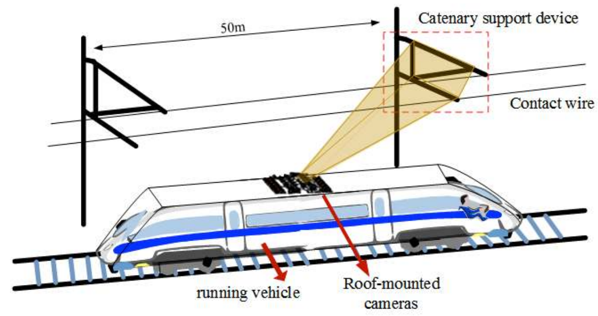


Fig. 2 Sketch map of the catenary support device image acquisition.

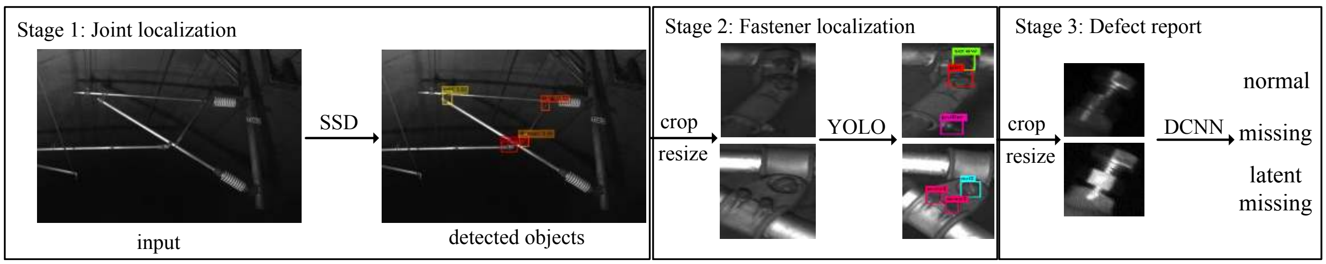


Fig. 3 The pipeline of the detection system that includes a three-stage cascaded DCNN.

A. Joint Localization

The goal of the first DCNN is to localize and extract the three-class cantilever joints in the captured catenary support device images. From different shooting angles, the object joints have multiple scales. To localize the joints in the captured HD images, SSD framework that performs well in both speed and accuracy is introduced. The input 6600×4400 pixels HD images are first resized to 660×440 pixels in order to alleviate the memory footprint of the model.

B. Fastener Localization

The output of Stage 1, namely, the extracted joints, is transmitted to Stage 2. In the extracted cantilever joints images, fasteners are relatively easy to distinguish since they cover a large area of the images and are usually not overlapped. Thus, a fast localization architecture based on the YOLO framework is cascaded in Stage 2.

C. Fastener State Classification and Defect Recognition

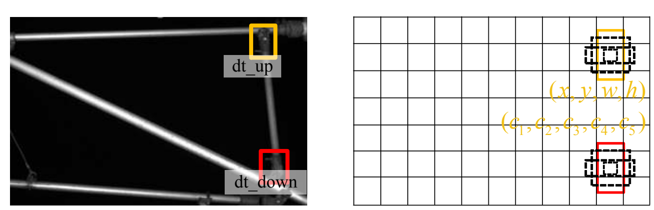
The extracted fasteners are classified into normal, missing and latent missing states based on the likelihood via a third DCNN. Fasteners are of small sizes and hence a lightweight DCNN is built to recognize defects in case of computation burden in Stage 3.

To be noted, since this paper focuses on the image processing of the captured catenary support device images, the details of the image acquisition steps will not be mentioned. In addition, the image processing-based detection is operated offline.

III. Detection Module

A. Localization of the Cantilever Joints Using SSD

The core idea of the SSD framework [22] is to produce a collection of default bounding boxes and predict the object class from the default boxes. As shown in Fig. 4, the default boxes are produced from the feature maps in different convolutional layers with different aspect ratios and scales. For a default box in the $m\times n$ feature map, the confidences of 5-class object including the background categories and the 4 indicators (x,y,w,h) that specify the regression box's coordinates are calculated. Each cell in a feature map can produce 4 default boxes by changing the ratio of the default box's length to width in the range of $\{\frac{1}{2},2\}$. Thus, the output of a convolutional layer is a tensor of $m\times n\times(4+5)\times4$.



(a) (b)

Fig. 4 Default box production of the SSD framework. (a) Input with ground truth boxes. (b) 11×7 feature map.

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, “Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network”. IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

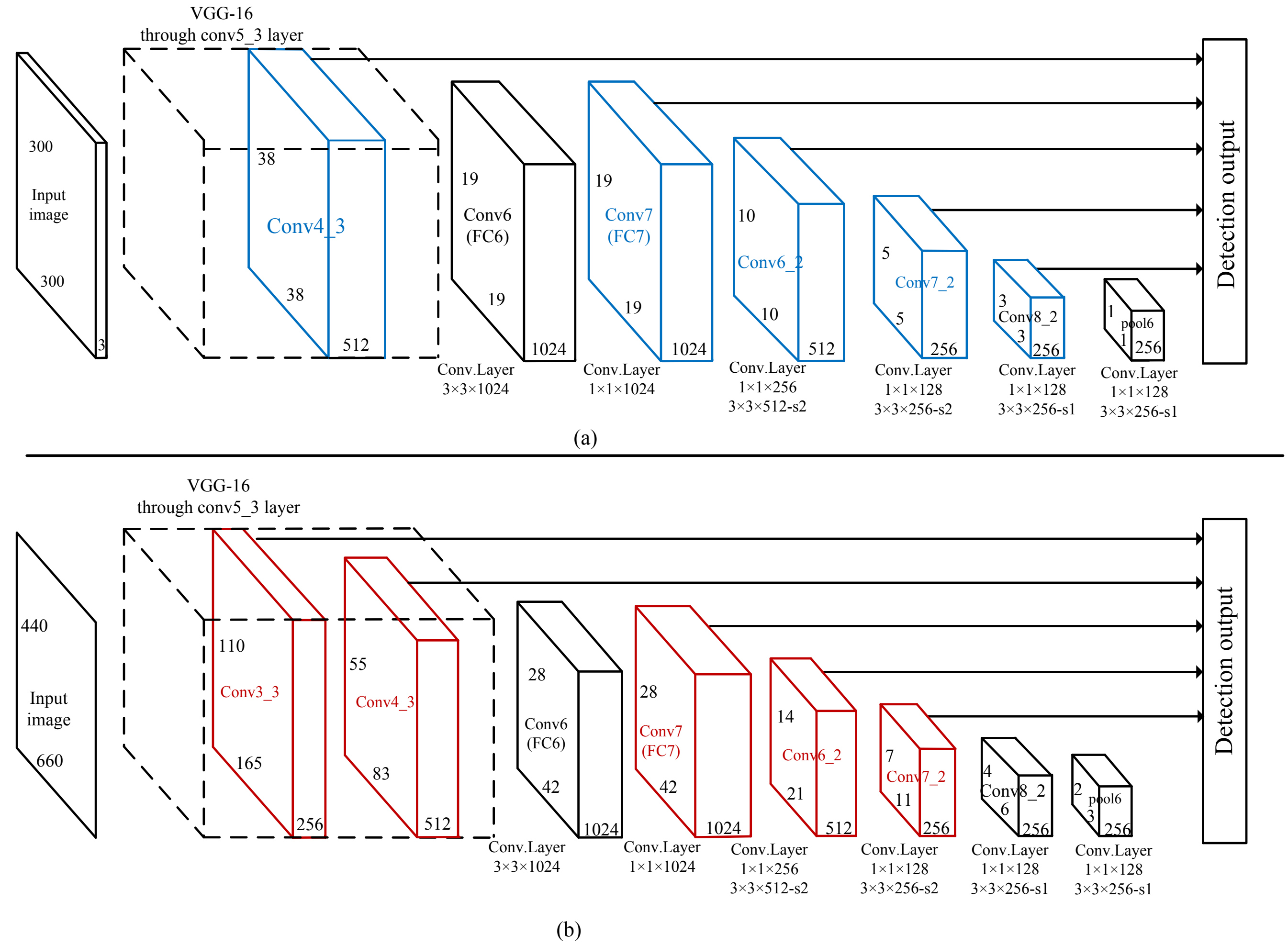


Fig. 5 DCNN architectures of SSD framework. (a) The SSD architectures in Ref [22]. (b) The modified architectures. The main optimization of the two SSD architectures is the different configurations for the detection layers.

In the original SSD architecture, based on VGG-16 network [30], *conv4_3, conv6, conv8_2, conv9_2, conv10_2, conv11_2* are selected as the output layers. Accordingly, the default boxes are produced on the multi-scale feature maps in sizes of 38×38, 19×19, 10×10, 5×5, and 3×3. Since the input images of Stage 1 consist of many small objects, the lower *conv3_3* is added to collect more low-level cues for detection (see Fig. 5). The size of input images is zoomed to 660×440 pixels. Thus, the modified SSD architecture includes the output layers *conv3_3*, *conv4_3*, *conv7*, *conv6_2*, *conv7_2*, *conv8_2* with feature maps at sizes of 165×110, 83×55, 42×28, 21×14 and 11×7.

Training Procedure:

For object localization problems, training data are comprised of the images and the ground truth boxes of each object. The key of the training process in SSD framework is to match the ground truth boxes to a series of fixed-size default boxes. The default boxes that overlaps the ground truth for more than 50% or the best overlapped default box are determined as the positives. A hard negative mining strategy picks the non-matched default boxes with high confidence as the negative training samples to balance the ratio of the positives to negatives in 1:3.

The object localization model is trained by minimizing a multi-task loss function (see Fig. 6) that sums the localization loss and the confidence loss. The localization loss is a Smooth L1 loss between the predicted box and the ground truth. The classification loss is a softmax class loss over the multiple classification confidences.

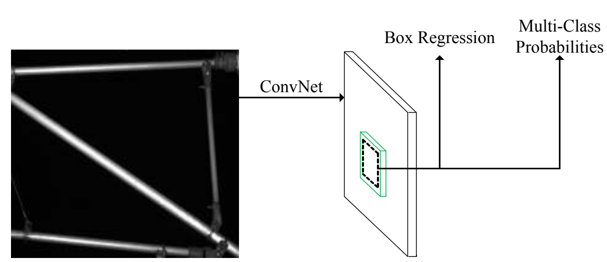


Fig. 6 Multi-task Loss of SSD.

To enhance the robustness of the proposed model, data augmentation, including random expansion, random crop and horizontal flip, is introduced to increase the training samples.

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

B. Localization of the Fasteners Using YOLO

As shown in Fig. 7, the core idea of the YOLO framework [21] is to predict multi-class bounding box candidates directly from the grids in the full input images. The combination of the class probabilities and bounding box confidence provides the resulting detection.

In Stage 2, the input images are divided into 7×7 grids. In the shooting angle, the nut and α-pin captured in front and reverse views are considered as different classes. Thus, each grid predicts classification probabilities for eight-class fasteners and two candidate bounding boxes with the confidence score. Each bounding box contains 5 position indicators, including the box coordinates (x, y, w, h) and the position confidence. Overall, the net's output is a tensor of $7\times7\times(2\times5+8)$.

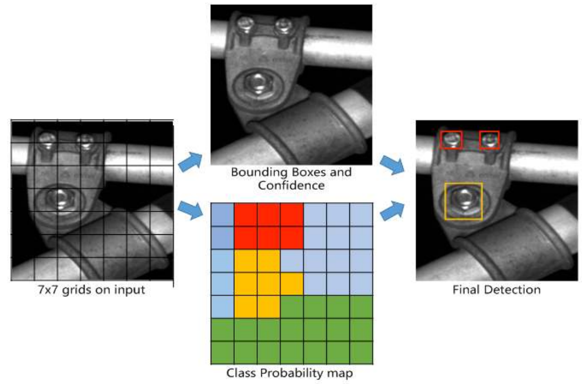


Fig. 7 YOLO framework.

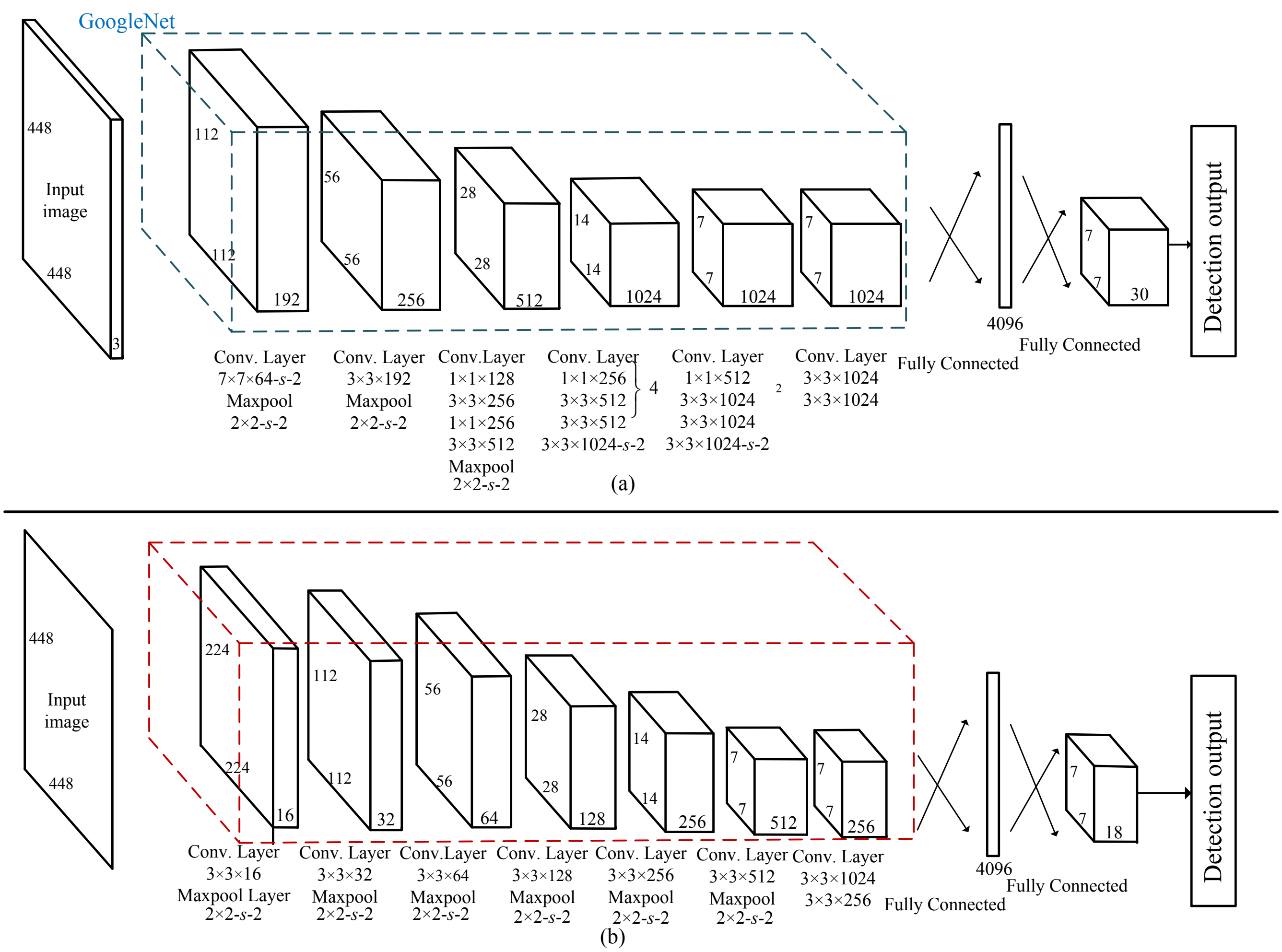


Fig. 8 DCNN architecture of YOLO framework. The original DCNN architecture (a) in [21] is based on the GoogleNet, while the architecture in this paper (b) is simplified to a light network.

Inspired by the GoogleNet [25], original YOLO network has 24 convolutional layers followed by two fully connected layers. Since the task in Stage 2 is relatively simple, a light YOLO architecture with 8 convolutional layers and two fully connected layers is introduced in Stage 2 as shown in Fig. 8.

Training Procedure:

The sizes of the joints are in the range of 300×300 to 600×600 pixels. To make good use of computational resources and to maintain the precise information of joints, the output of Stage 1 is resized to 448×448 pixels.

The training loss of Stage 2 is based on sum-squared error and comprised of five parts, i.e., the regression-weighted

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

sum-squared error of each cell's bounding box center x and y, the square root of each bounding box width and height, the sum-squared error of the saliency probability of whether objects exist in a bounding box, the classification-weighted sum-squared error of the saliency probability of whether an object does not exist in a bounding box, and the class probabilities of each cell. Dropout and random crops are introduced to reduce overfitting.

C. Defect Judgment of the Fasteners

The fasteners include three basic states: the normal working, missing and latent missing states. The goal of Stage 3 is to categorize the extracted fasteners into three classes and correspondingly recognize the defect states. Fig. 9 lists the states for each type of fastener. For the nut and α-pin, a defect cannot be judged on the reverse side images. Hence the extracted nut and α-pin in reverse side will not be input into Stage 3.

For some of the missing states, the fasteners cannot be localized in Stage 2. Since the number and class of the fasteners in these joints are fixed, the defect can be judged by the absence of the fasteners in Stage 2, as shown in Fig. 10. In addition, an image classification network is built to categorize the installation states. The architecture of the state classification network is summarized in Fig. 11. It contains a total of four convolutional layers and two fully connected layers between the input and output layer.

To unify the training process, the output layer is connected to a 16-way softmax that produces the probabilities for 16-class fastener states. This network will provide a probability for the states that the fasteners belong to and judge the states by a threshold.

Training Procedure:

The fasteners are tiny objects, with sizes of approximately 70×70 pixels. Due to the limited samples of defect images,

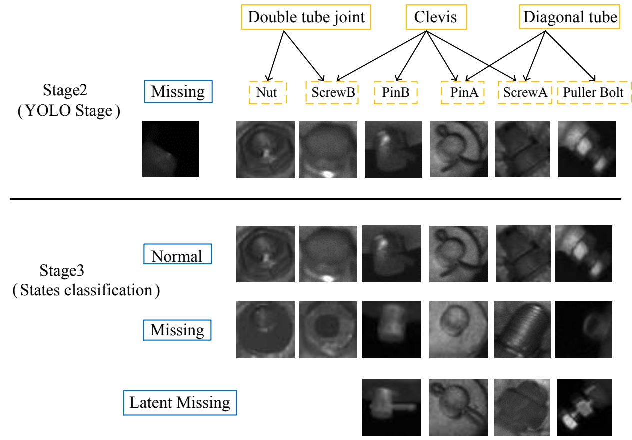


Fig. 9 Categories for the defects of fasteners in Stage 3. The loosening of the screws and puller bolts and the open lack of the pins are defined as latent missing.

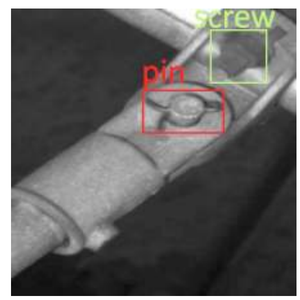


Fig. 10 Detection of the missing puller bolt by YOLO in Stage 2.

data augmentation is introduced. For the sake of balancing the training, the number of the normal input samples is limited to balance the defective samples.

In Stage 3, the training loss is no longer multi-task. Since it is a multi-label classification problem, softmax class loss is also used to compute the confidence of the classification. Meanwhile, dropout is also adopted here by 50% at *conv5* layer to reduce overfitting.

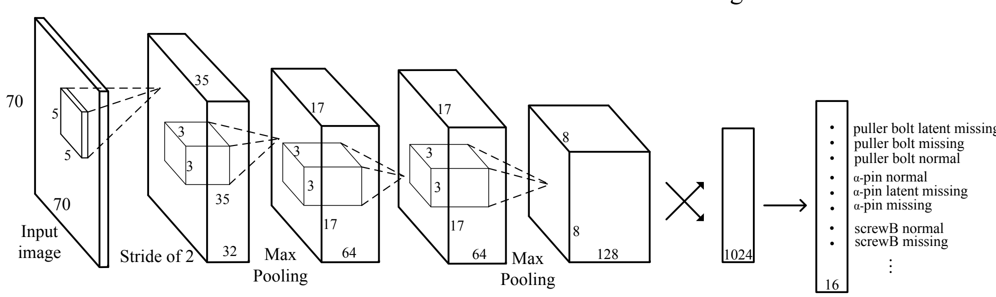


Fig. 11 DCNN architecture of the fastener states classification.

IV. Experiment and Results

The above analysis of the proposed cascaded detection system provides the feasibility to automatically localize the cantilever joints of the catenary support device and recognize the defects of their fasteners.

A. Dataset

The dataset used in the experiments consists of the catenary support device images captured from an approximately 100-kilometer line along the Ju-Yue section of the Wuhan-Guangzhou high-speed railway, in which 2000 catenary support devices and 40000 fasteners exist. The images are collected by the XLN4C-01 imaging inspection vehicle (see Fig. 12) during the night. The dataset contains the catenary support devices in various challenging environments, such as tunnels, turnouts and viaducts, to evaluate the robustness of the proposed method.

To build the training set for Stage 1, we manually draw the bounding boxes and assign the labels of approximately 8,563

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

Fig. 12 XLN4C-01 inspection vehicle.

catenary support device images, in which 6,371 images are in the training set and 2,192 images are in the validation set.

The training loss guides the training process and the accuracy indicates the reliability of the trained model. To avoid overfitting, the validation set is built to choose the trained model. The accuracy of the validation set is calculated in a defined interval, and the model with the highest accuracy is chosen as the testing model.

A testing dataset is generated to evaluate the proposed method. To prove the adaptability of the model, the testing dataset consists of the images collected from a different section Heng-Zhu, 67 km in total. In total, the testing dataset consists of 4,487 images.

B. Training Process

The dataset is employed to validate the proposed system. The experimental environment is described as follows: Deep learning open source framework Caffe [31], Ubuntu 14.04, 32GB RAM, CPU clocked at 3.2 Hz, and GTX 1080 GPU with 8GB memory.

The joints on the 6,371 images in the training set are manually labeled. Since the task of Stage 2 is much simpler, Stage 2 can be considered as a semi-supervised training. Only 1500 cantilever joint images are manually labeled. Then when the model trained by 1500 images reaches an adequate accuracy, it is used to detect and generate the label information of the rest of cantilever joint images in the training set. The training label generation and training are alternatively conducted when the label information of 2000 images are generated.

Due to the limitations of the defect fastener images, the training dataset in Stage 3 contains 35 images for each defect state and normal states of six-class fasteners, for a total of 560 images for 16 types of states. With the data augmentation strategy, the training datasets are expanded.

Each of the three DCNN stages is end-to-end trained using back-propagation and stochastic gradient descent (SGD) solver with momentum (0.9) and weight decay of 5×10^{-4}. Due to the limitation of GPU memory and training samples, the training batch sizes of the three stages are set to 16, 8 and 1. The learning rate is used to control the rate of gradient descent of the training loss. In Stage 1, the learning rate is set to 0.0001 initially and then tuned to 0.001 after 4000 iterations. Stage 2 and Stage 3 employ fixed learning rates of 0.0005 and 0.01.

C. Experiment Result and Discussion

The testing images are collected from a different route but are tested under the same computation environment as training. The proposed method displays good results in localizing the joints and fasteners, and recognizing their defects in the three stages. Fig. 13 shows several visualized detection examples and results.

To verify the effectiveness of the proposed method, three sets of experiments are conducted to evaluate the method in terms of the average precision and the processing time costs (frames per second, FPS), including the effects of the modified DCNN structures, the comparison with other underlying DCNN architectures and shallow learning algorithms and the effectiveness of the three-stage cascade architecture.

The true positive (TP), false negative (FN) and false positive (FP) are counted to compute the following statistical indicators precision and recall. The mAP (mean average precision) is computed according to the relationship P(R) of precision (P) and recall (R).

$$\text{Precision} = \frac{TP}{TP+FP}\times 100\% \tag{1}$$

$$\text{Recall} = \frac{TP}{TP+FN}\times 100\% \tag{2}$$

$$\text{mAP} = \int_0^1 P(R)dR \tag{3}$$

To be noted, the evaluation of effects of the modified DCNN architectures is designed to prove that the DCNN architecture is correctly selected and modified for each stage. To give a fair evaluation, the experiment of Stage 2 leverages the human verified outputs of Stage 1 thus the false positive outputs will be ignored. Moreover, some of the joints are severely occluded (See Fig. 14). The missing of these components will not be counted when calculating precision.

1) Effects of the modified DCNN architectures:

Accuracy of using multiple output layers in Stage 1:

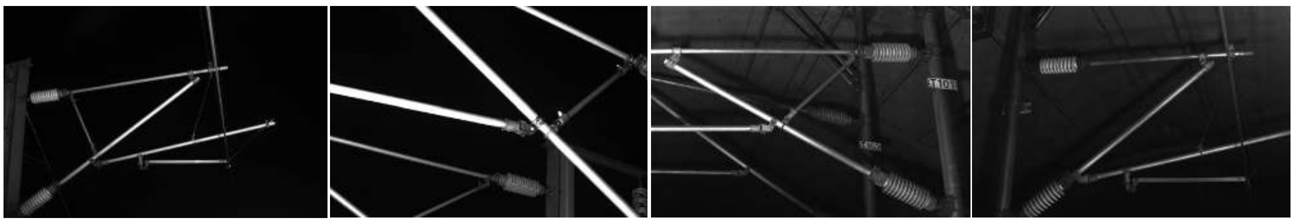
The modified architecture for SSD framework is compared with the original SSD architecture to analyze the effects of using multiple output convolutional layers. TABLE I shows the detection results of different configurations of the output layers. The performance is increased when the outputs are predicted from multiple layers. The comparison of the proposed model and the original model shows that accuracy is improved when using the low-level convolutional layers. This is not surprising since the proposed system consists of many small objects. Pruning *conv8_2* at output will also improve the accuracy because the feature maps in this layer contain very coarse information.

The proposed architecture is based on VGG-16 and is also compared with that based on Resnet-50 in the experiments. Since 2015, the residual network has been very successful in the ImageNet classification. The results are summarized in TABLE II. For each of the joints, the detection result is of low accuracy on the validation set and shows it is overfitting. This is not surprising since the number of the training data cannot satisfy training a Resnet-50 and since the objects are not as complex as the ImageNet.

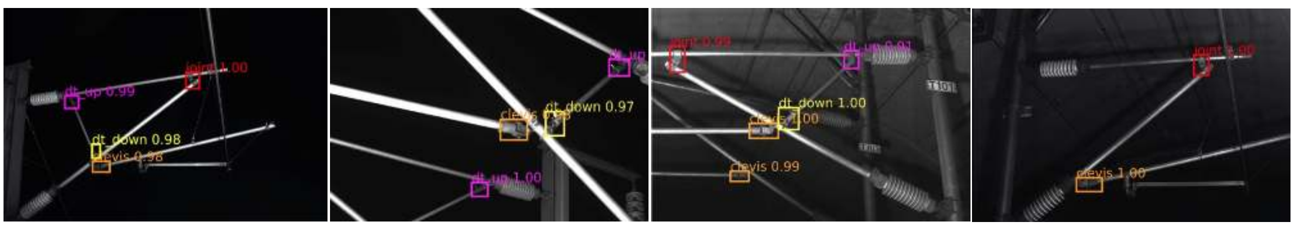
Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

TABLE I EFFECTS OF DIFFERENT OUTPUT LAYER OPTIONS

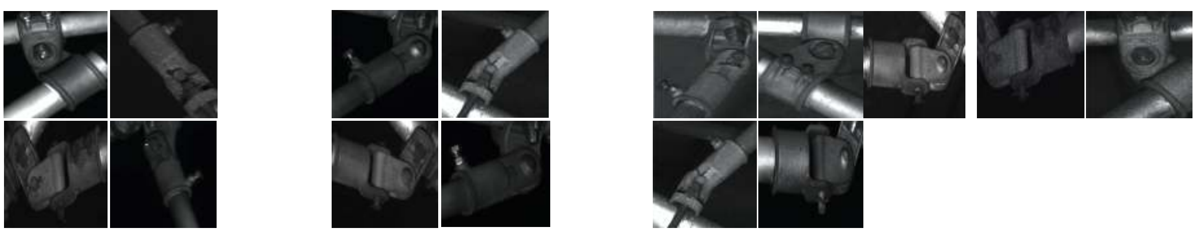
Configuration	Prediction from the following layers						mAP
	conv3_3	conv4_3	conv7	conv6_2	conv7_2	conv8_2	
Proposed SSD	√	√	√	√	√		92.16
Original SSD[20]		√	√	√	√		82.73
Alternative 1	√	√	√	√	√	√	88.78
Alternative 2		√	√	√	√		79.26
Alternative 3		√	√				70.52



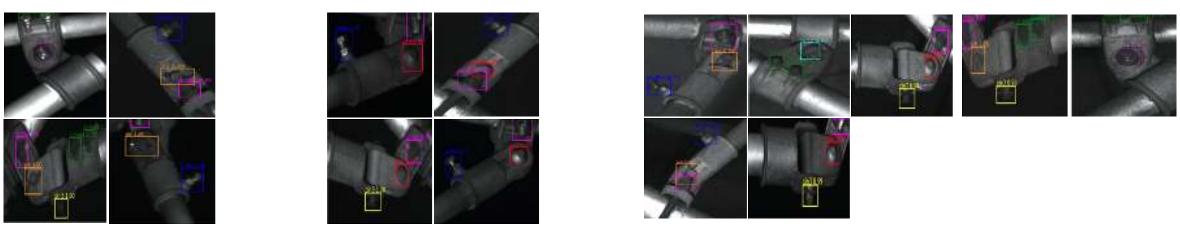
(a)



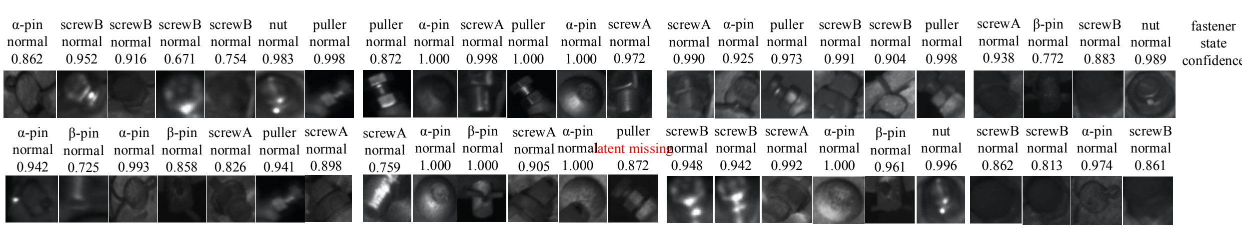
(b)



(c)



(d)



(e)

Fig. 13 Four examples of defect detection. (a). The captured catenary support device images. (b). Cantilever joints localization using the SSD framework. (c). Crop and resize of the cantilever joints. (d). Fasteners localization using the YOLO framework. (e). Fastener state classification. The four examples contain the fasteners all in the normal states.

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

(a) (b)

Fig. 14 The severe occluded joints.

TABLE II THE BASIC NETWORK COMPARISON OF STAGE 1

network	Resnet-50	VGG-16
mAP	17.19	89.16
dt up	12.01	87.88
dt down	17.5	83.21
joint	22.24	92.16
clevis	23.91	91.74

Running efficiency of the light DCNN architecture in Stage 2:

To evaluate the running efficiency of the light network, we compare it with the original YOLO architecture under the same environment and the results are listed in TABLE III. It can be seen that both of the DCNN architectures have good performance, but the proposed light YOLO offers a speed up. It should be noted that experiments are processed using GTX1080. The improvement of GPU will accelerate the model by a large margin.

TABLE III COMPARISON TO THE ORIGINAL YOLO

method	mAP	FPS	Training time consumption
The proposed fast architecture	96.72	83	133 min
Original YOLO network	96.85	12	251 min

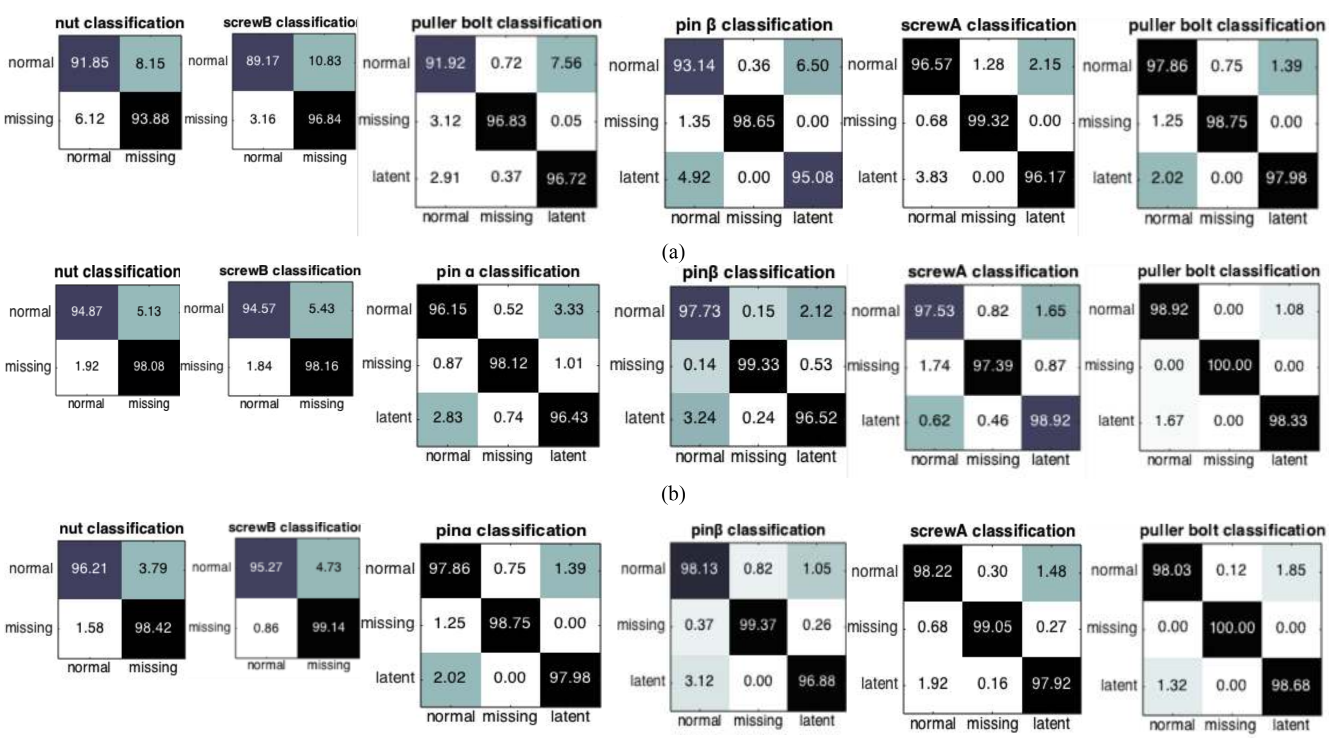
Comparison of the multiple DCNN architecture in Stage 3:

The proposed DCNN architecture in Stage 3 is compared with a light architecture that refers to a vehicle logo recognition system [33] and a large architecture AlexNet [24]. The light network contains two convolutional layers, two pooling layers and a fully connected layer to classify 11-class logos. AlexNet contains five convolutional layers, three max-pooling layers and three fully connected layers to classify 1000-class objects in the ImageNet competition. In the comparative experiment, the three DCNNs are trained and tested using the same dataset.

Confusion matrices are used to evaluate the classification accuracy. It can be seen from Fig. 15 that the proposed system and AlexNet perform better on accuracy than the light network. However, according to TABLE IV, the large network decreases the speed of the task compared to the proposed network.

TABLE IV COMPARISON OF THE THREE SCALE CLASSIFICATION NETWORK

Method	mAP	FPS	Training time consumption
light	83.64	634	55 min
medium	94.72	420	74 min
large	94.88	83	93 min



(c)

Fig. 15 Confusion matrices of fastener state classification. (a), (b), (c) use the light, medium and large network respectively. The rows of the confusion matrices respond to the correct class while the columns display the predicted classes. Latent missing is simplified as "latent".

According to the confusion matrices, for the latent missing states, the proposed classifier shows a relatively higher accuracy on the puller bolts and two screws. However, some of the fasteners in normal states are considered as latent missing (see Fig. 13 (e)). The latent missing of fasteners sometimes appears as the normal state and it is a close call. Since the most important task is to prevent the absence of defect recognition, a small amount of the false prediction of the normal as defects is allowed to some extent.

2) Comparison with other underlying DCNN architectures and shallow learning algorithms:

The proposed localization and classification networks are compared with the following learning algorithms considering accuracy and speed.

Localization Comparison:

- **SSD architecture**: SSD framework has been explained in Section III.A.
- **YOLO architecture**: YOLO framework has been explained in Section III.B.
- **Faster R-CNN:** Based on ZF net, the *conv_5* feature is input into a region proposal network to generate ~2K region proposals. Then, the region proposals are reflected to the *conv_5* and are classified by two fully connected layers and a softmax. Finally, the predicted bounding boxes are slightly adjusted to fit the objects.
- **HOG features with AdaBoost classifiers**: Histogram of gradient [23] is a local hand-craft feature descriptor that is invariant to light and rotation. Object detection is achieved by sliding windows on the input images. For each window, the HOG feature is calculated and then classified by a series of cascaded two-category classifiers. The classifier is trained by an AdaBoost algorithm [34] that highly weights the wrong prediction in the previous classifier by an adaptive boost training mechanism.
- **Deformable Part Models (DPM)**: DPM [35] is also based on HOG features but it calculates multi-scale pyramid features of the input images. Objects are modeled by the part and root filters in coarse-to-fine resolution. A latent SVM is used to train the part models and are combined with a margin-sensitive approach for data mining hard negatives.

A precision-recall (PR) curve is drawn to visualize the performance for different detection algorithms. It can be seen from Fig. 16 that the SSD architecture and Faster R-CNN perform better than YOLO and DPM in terms of accuracy.

The statistical results are summarized in TABLE V. In particular, the SSD and Faster R-CNN have relatively higher accuracy in Stage 1 than the others. However, the SSD network runs 3x faster than the Faster R-CNN.

Since the proposed system has a great capacity of HD images to process, the SSD framework should be accepted as the extractor of the cantilever joints. In Stage 2, the proposed deep learning algorithms (including SSD, YOLO and Faster R-CNN), and even DPM, have good performances in accuracy since Stage 2 is not as complex as Stage 1. However, the proposed fast YOLO network has a huge superiority in detection speed and training time consumption.

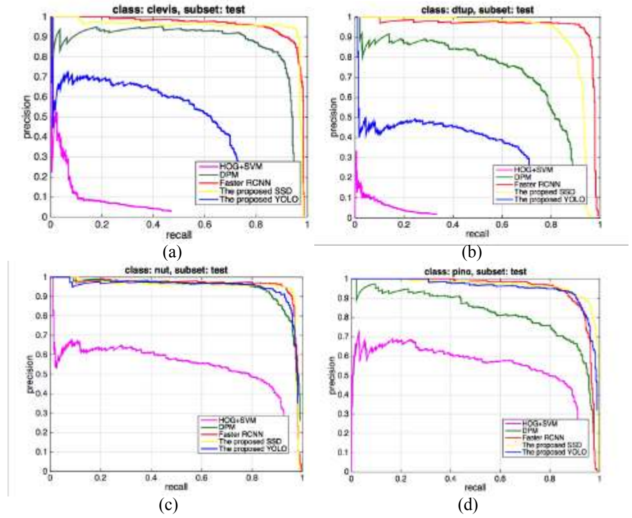


Fig. 16 PR curves for the localization results. (a) (b) plot the result of the clevis and the upper diagonal tube in Stage 1. (c) (d) plot that of the nut and the ά-pin in Stage 2.

TABLE V Comparison of the Joints and Fasteners Extraction Results

method	mAP of Stage 1	mAP of Stage 2	detection FPS of Stage 1	training time consumption of Stage 1
SSD framework	92.16	97.41	12	107 h
YOLO framework	74.32	95.56	84	31 h
Faster R-CNN	90.03	96.24	4	132 h
DPM	80.51	94.69	0.47	124 h (Input size: 200 images)
HOG+AdaBoost	57.92	73.28	1.42	82 h (Input size: 200 images)

Defect recognition comparison:

In addition to the proposed methods of object localization, DCNN in Stage 3 is also compared with several image classification methods.

- **The proposed DCNN architecture**: The proposed architecture has been explained in Section III.C.
- **HOG features with SVM**: The input images are calculated of HOG features and classified by the two-category classifier SVM [36].
- **SIFT features with template matching**: Scale-invariant feature transform [37] is a local feature descriptor that calculates the interest points in multi-scale space and collects the key interest points of the two images to be matched.

The comparative results are summarized in TABLE VI. Apparently, the proposed DCNN-based method outperforms the shallow learning in both accuracy and speed.

TABLE VI COMPARISON OF THE DEFECTS RECOGNITION RESULTS

Method	mAP	FPS	Training time consumption
The proposed DCNN	92.78	636	55 min
HOG+SVM	71.66	12	62 min
SIFT+template matching	65.32	59	

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345

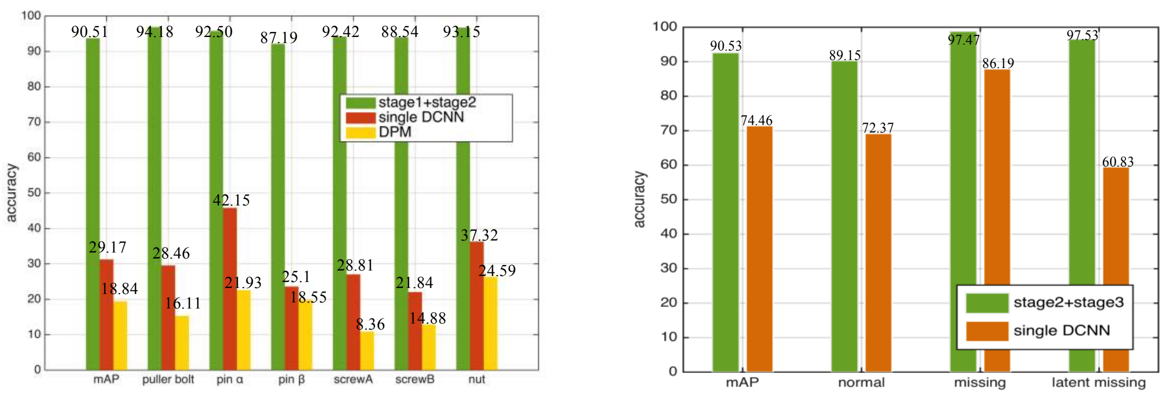
3) Effectiveness of the three-stage cascade architecture:

To investigate the impact of the three-stage cascade architecture, two comparative experiments are conducted by the combination of Stage 1, 2 and Stage 2, 3.

Stage 1 and Stage 2 are combined into a single DCNN that can directly localize the tiny fasteners in the captured HD images. The proposed SSD framework in Stage 1 is used as the single DCNN in the comparative experiment. Fig. 17 (a) lists the result of localizing the six-class fasteners. Unfortunately, the single DCNN shows very poor performance in terms of accuracy. This is not surprising as it is difficult to distinguish the 20×20 pixels objects in the 6600×4400 pixels raw input images. Even *conv3_3*'s receptive field is still too large to predict the tiny fasteners. The low-level layers such as *conv2_3* have less semantic information about the objects, which does not help in detecting small objects. Due to the limited computation resource, the re-scaling of the input also eliminates the precise information of tiny fasteners.

Owing to the disadvantage of the DCNN architecture caused by the large receptive fields, one of the best shallow learning DPM replaces the single DCNN to be compared with the two-stage DCNN. The shallow learning algorithm uses the sliding windows on the raw input images. The size of the sliding windows can be adjusted to fit the objects. However, DPM also shows a low performance because the hand-craft feature is not as robust as the feature learning, especially for the fasteners with simple structures. Overall, it is better to localize the joints and the fasteners separately in two stages.

Stage 2 and Stage 3 are also combined to output the states of the fasteners in the extracted joints. The defective states of fasteners are labeled in the joint images to train a DCNN architecture based on the YOLO framework in Stage 2. To balance the training, 35 images are selected for each type of fastener in all states. However, the absence of adequate defect samples and the minor discrepancy between the normal and defective states can result in poor accuracy. Fig. 17 (b) lists the mAP for the three states and shows that the normal and latent missing states are easily missed using the single network.



(a) (b)

Fig. 17 Effect analysis of the cascade DCNN structure. (a) Performance comparison of the two-stage and unified DCNN for detecting the fasteners. (b) Performance comparison of the two-stage and the unified DCNN in the recognition of defects on the joint images.

Overall, the cascaded three-stage DCNN is necessary to accurately recognize the defect states of tiny fasteners from the HD catenary support devices.

V. Conclusion

This paper presents a method to detect the defective fasteners of the cantilever joints on the catenary support devices. The proposed three-stage architecture can automatically localize the three cantilever joints and the six fasteners and judge the missing and latent missing states of the fasteners in the captured images. All stages are accomplished by deep convolutional neural networks, which benefit the detection due to the superiority in robustness and adaptability. Overall, the proposed approach shows a promising application and accuracy in the fasteners' defect recognition. The reduced time consumption makes it feasible to periodically detect the enormous quantity of the catenary fasteners in a large railway network. Nevertheless, the results suggest some further improvements.

(1) The catenary support device has many more items than the defective fasteners to detect, such as cracks on the joints, the loosening of bracing wires and the flashover of insulators.

(2) The latent missing of nut and screw B cannot be judged due to the blind angle of cameras. Thus, detection based on RGB-D data can be attempted to build a DCNN-based tridimensional model to address it.

VI. Acknowledgement

The authors would like to thank the Guangzhou Railway Group for providing the experiment images and the anonymous reviewers for their valuable comments.

References

[1] S. Bruni, J. Ambrosio, A. Carnicero, et al., "The results of the pantograph–catenary interaction benchmark," *Vehicle Syst. Dyn., Int. J. Vehicle Mech. Mobility,* vol. 53, no. 3, pp. 412-435, Nov. 2014.

[2] Fittings for Overhead Contact System in Electrification Railway, China Railway Industrial Standard, TB/T 2075.1, 2010.

[3] Z. Liu, W. Liu, Z. Han, "A high-precision detection approach for catenary geometry parameters of electrical railway," *IEEE Transactions*

on Instrumentation and Measurement, 2017, DOI 10.1109/ TIM.2017.2666358.

[4] C. J. Cho, H. Ko, "Video-based dynamic stagger measurement of railway overhead power lines using rotation-invariant feature matching," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, no.3, pp. 1294-1304, Oct. 2015.

[5] E. Karakose, M. T. Gencoglu, M. Karakose, et al., "A new experimental approach using image processing based tracking for an efficient fault diagnosis in pantograph-catenary systems," *IEEE Trans. Ind. Inform.*, 2016, DOI 10.1109/ TII.2016.2628042.

[6] Y. Han, Z. Liu, D. J. Lee, et al., "High-speed railway rod-insulator detection using segment clustering and deformable part models," in *IEEE International Conference on Image Processing*, 2016, pp. 3852-3856.

[7] Q. Li, S. Ren, "A real-time visual detection system for discrete surface defects of rail heads," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no.8, pp. 2189-2199, Feb. 2012.

[8] H. Feng, Z. Jiang, F. Xie, P. Yang, et al., "Automatic fastener classification and defect detection in vision-based railway detection systems," *IEEE Trans. Instrumentation and Measurement*, vol. 63, no. 4, pp. 877–888, Apr. 2014.

[9] F. Marino, A. Distante, P. L. Mazzeo, et al., "A real-time visual detection system for railway maintenance: automatic hexagonal-headed bolts detection," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 3, pp. 418-428, Apr. 2007.

[10] C. Aytekin, Y. Rezaeitabar, S. Dogru, et al., "Railway fastener inspection by real-time machine vision," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 7, pp. 1101–1107, Jul. 2015.

[11] Y. Lecun, Y. Bengio, G Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-44, May. 2015.

[12] X. Gibert, V. M. Patel, and R. Chellappa, "Deep multitask learning for railway track detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–12, Jul. 2016.

[13] L. Oneto, E. Fumeo, B. Clerico, et al., "Dynamic delay predictions for large-Scale railway networks: deep and shallow extreme learning machines tuned via Thresholdout." *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. PP, no. 99, pp. 1-14, 2017.

[14] L. Oneto, E. Fumeo, B. Clerico, et al., "Innovative applications of big data in the railway industry.", IGI Global, 2017.

[15] H. Li *et al.*, "Improving rail network velocity: A machine learning approach to predictive maintenance," *Transp. Res. C Emerg. Technol.*, vol. 45, pp. 17–26, 2014.

[16] X. Gibert, V. M. Patel, and R. Chellappa. "Material classification and semantic segmentation of railway track images with deep convolutional neural networks," in *IEEE International Conference on Image Processing*, 2015.

[17] M. Everingham, L. Van. Gool, C. K. I. Williams, et al., "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303-338, Jun. 2010.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[19] R. Girshick, "Fast R-CNN," in *IEEE International Conference on Computer Vision*, 2015, pp. 1440-1448.

[20] S. Ren, K. He, R. Girshick, et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1-1, Jun. 2016.

[21] J. Redmon, S. Divvala, R. Girshick, et al., "You only look once: unified, real-time object detection," in *IEEE Conference on Computer Vision and Pattern Recognition,* 2016, pp. 779-788.

[22] W. Liu, D. Anguelov, D. Erhan, et al., "SSD: Single shot multiBox detector," *arXiv preprint arXiv: 1512.02325*, 2015.

[23] N. Dalal, B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[24] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, no. 2, Dec. 2012.

[25] C. Szegedy, W. Liu, Y. Jia, et al., "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1-9.

[26] Y. Tang, X. Wu, "Scene text detection and segmentation based on cascaded convolution neural networks," *IEEE Transactions on Image Processing*, vol. pp, no.99, Jan. 2017.

[27] Y. Huang, X. Liu, L. Jin, et al, "DeepFinger: A cascade convolutional neuron network approach to finger key point detection in egocentric vision with mobile camera," in *IEEE International Conference on Systems, Man, and Cybernetics*, 2015, pp. 2944-2949.

[28] Y. Sun, X. Wang, X. Tang, "Deep convolutional network cascade for facial point detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3476-3483.

[29] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multi-task cascaded convolutional networks," *IEEE Signal Processing Letters,* vol. 23, no. 10, pp. 1499-1503, Oct. 2016.

[30] K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[31] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093,* 2014.

[32] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[33] Y. Huang, R. Wu, Y. Sun, et al., "Vehicle logo recognition system based on convolutional neural networks with a pretraining strategy," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, no.4, pp.1951-1960, Feb. 2015.

[34] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* 2001, pp. 511–518.

[35] P. Felzenszwalb, D. Mcallester, D. Ramanan. "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1-8.

[36] C. Cortes, V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no.3, pp. 273-297, Sep. 1995.

[37] D. G. LOWE. "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no.2, pp. 91-110, Nov. 2004.

Junwen Chen (S'17) received a B.S. degree in electrical engineering and automation from Southwest Jiaotong University, Chengdu, China, in 2015. She is currently working towards an M.Sc. degree at Southwest Jiaotong University, with a focus on the detection and diagnosis of the railway pantograph-catenary system. Her work lies in image processing, machine learning, computer vision and their application in the detection.

Zhigang Liu (M'06–SM'16) received a Ph.D. degree in power systems and automation from Southwest Jiaotong University, Chengdu, China, in 2003.

He is currently a full professor with the School of Electrical Engineering, Southwest Jiaotong University. His current research interests include electrical relationships of vehicle grids in high-speed railways, power quality considering grid connections of new energies, pantograph-catenary dynamics, fault detection, status assessment and active control.

Hongrui Wang (S'15) received a B.S. degree in electrical engineering and automation from Mao Yisheng Class, Southwest Jiaotong University, Chengdu, China, in 2012, where he is currently pursuing a Ph.D. degree with the School of Electrical Engineering.

He is involved in a Joint Ph.D. Program by the China Scholarship Council with the Section of Railway Engineering, Delft University of Technology, Delft, The Netherlands. His current research interests include signal decomposition, time-frequency analysis, signal filtering, and machine learning and their applications in the assessment, detection, diagnosis, and maintenance of railway pantograph-catenary system.

Alfredo Núñez (M'02–SM'14) received a Ph.D. degree in electrical engineering from the Universidad de Chile, Santiago, Chile, in 2010.

He was a post-doctoral researcher with the Delft Center for Systems and Control, Delft, The Netherlands. He is currently with the Section of Railway Engineering, Delft University of Technology, Delft. He has authored a book titled Hybrid Predictive Control for Dynamic Transport Problems in the Series of Advances in Industrial Control (Springer-Verlag, 2013). His current research interests include monitoring and maintenance of railway infrastructure, modeling and control of traffic and transportation systems, model predictive control, and fuzzy systems.

Zhiwei Han (M'16) received a Ph.D. degree in power systems and automation from Southwest Jiaotong University of China in 2013.

He is currently a lecturer with the School of Electrical Engineering, Southwest Jiaotong University. His current research interests include modern signal processing, computer vision, and their application in railway and electric power system.

Please cite as: J. Chen, Z. Liu, H. Wang, A. Núñez, and Z. Han, "Automatic defect detection of fasteners on the catenary support device using deep convolutional neural network". IEEE Transactions on Instrumentation and Measurement 67(2): 257-269, February 2018. DOI: 10.1109/TIM.2017.2775345