Rail Surface Defect Detection and Severity Analysis using CNNs on Camera and Axle Box Acceleration Data

Kanwal Jahan
German Aerospace Center (DLR)
Lilienthalplatz 7
38108 Brunswick, Germany
kanwal.jahan@dlr.de

Alexander Lähns
German Aerospace Center (DLR)
Rutherfordstr. 2
12489 Berlin, Germany

Benjamin Baasch
German Aerospace Center (DLR)
Rutherfordstr. 2
12489 Berlin, Germany
benjamin.baasch@dlr.de

Judith Heusel
German Aerospace Center (DLR)
Lilienthalplatz 7
38108 Brunswick, Germany
judith.heusel@dlr.de

Michael Roth
German Aerospace Center (DLR)
Lilienthalplatz 7
38108 Brunswick, Germany
m.roth@dlr.de

ABSTRACT

Rail surface defect detection is a relevant problem in the field of data-driven railway maintenance. Artificial intelligence and neural networks (NN) for axle box acceleration (ABA) or camera data show great potential for defect detection and classification. However, a sufficient amount of labeled training data is required, all the more if the defect severity is to be estimated. A unique dataset of time-synchronized ABA and camera data is employed that contains labeled defect instances. For the image analysis, RetinaNet as a single-stage object detector (with the backbone of ResNet-50 and a feature pyramid network) is used to achieve high classification performance for the two most common rail surface defects (squat and corrugation). Additionally, a machine learning-based method on ABA data to estimate defect severity levels (low, medium, heavy) is proposed. False positives are detected in the original labels by both classifiers during evaluation. The inspection of the false positives in image data reveals that defects have been overlooked in the initial labeling. The insights of this work help to reduce the dependency on labeled data by using only a few labeled samples and by exploiting complementary data sources instead of increasing the number of labeled instances.

Keywords

Deep learning, Convolutional neural network, Time-synchronized dataset, Supervised learning, Severity analysis, Rail surface defects, Squats, Corrugation

1. INTRODUCTION

Knowing the condition of the track infrastructure is indispensable for cost-efficient maintenance planning and maintaining the safety standards of the railway network. A defective infrastructure can appear due to multiple reasons, e.g., violation of track geometry parameter specifications or the presence of short wavelength rail surface defects such as squats, head checks, or corrugation. The permanent presence of short wavelength defects is safety-relevant, can cause secondary defects, and reduces the passenger's comfort by vehicle vibrations and noise [14]. In extreme cases, the effects can be serious destruction of the property and the injury or death of crews and passengers [19].

Earlier, the detection of rail defects was usually conducted by manual inspection by experts, which is expensive and time-consuming for advanced high-speed railway systems. Visual inspection can fail to capture or register all defects [19]. Ever since efforts are made to include diverse sensors and detection methods. Nowadays, it is a common practice to use measurement trains on main rail lines to assess the track geometry parameters and to detect rail irregularities, using optical, ultrasound, vibration, and eddy-current sensors. These measurement trains usually operate on the same infrastructure every few months. Employing onboard sensor systems on regular trains which are capable of delivering similar information as measurement trains would allow for frequent but cost-efficient track inspections which enables degradation monitoring and early detection of defects.

Mostly, signal processing-based methods have been studied for vibration sensors such as axle box acceleration (ABA) sensors to provide track geometry parameters, [21] and to detect defects such as squats, corrugation [5], and insulated joints [10, 16]. On the other hand, conventional image processing has been deployed for the analysis of collected camera data replacing the physical visual inspection [15, 18]. Such methods require feature engineering, longer processing time per image, and the accuracy of the approach is domain specific. However, convolutional neural networks (CNNs) perform better on the mentioned criteria and can be retrained using custom datasets. Recently the inclusion of deep learning-based methods using camera images, for rail defect detection as well, has seen a rise [3, 7, 9]. However, the application of artificial intelligence (AI) using multiple sensors is not yet widespread in the railway domain but holds great potential, e.g., in the analysis of huge amounts of sensor data for automatic defect detection from vehicle-borne image and vibration data. So far approaches either use only image data-based detection or rely only on ABA-based classifiers while labeled images are used for validation only [6].

We propose methods for the detection of the defects on the rails and their further classification into severity levels from ABA data and for categorizing the defects into their types (corrugation and squats) based on image data. All approaches use CNNs for their respective classification. The results of the proposed network architectures are shown for a time-synchronized data set comprising both ABA and image data. From the maintenance perspective, the results obtained from the two data sources can be combined to obtain more detailed and reliable results. The novelty of this work is the possibility to

complement the knowledge obtained from different sensor sources i.e. ABA and camera.
The outline of this paper is as follows. Sec. 2. introduces the utilized data set, Sec. 3. explains the network architectures for the ABA data-based classifier (Sec. 3.1), and the image classifier (Sec. 3.2). Sec. 4. shows the results of the proposed methods applied to real data. Concluding remarks are given in Sec. 5.

2. Data Collection and Description

In the scope of this paper, the data-driven condition monitoring is performed on real sensor data collected from the measurement train equipped with multiple sensors which run on an existing railway network in the Netherlands. The measurement train captures the rail images by using high-definition line scan cameras with the raw data from other sensors, such as three-component ABA, and Global Navigation Satellite System (GNSS) receivers [1]. The highlight of this research is that the data collected from the camera and ABA is not only georeferenced but also time-synchronized, allowing us to combine the results. There are 6 cameras installed on each side of the vehicle, left and right, facing the rails directly. The image data used in this work is only from one left and one right camera. A part of the collected video data has been labeled by the experts manually, to be able to validate and assess the performance of the algorithms and classifiers for ABA data mainly. As the labeled camera images are available, may they be limited, they are used for training and evaluating an independent image-based classifier with the main focus of reducing manual labeling efforts in this research. Labels belonging to 5 different types include defects like head check Fig. 1(a), stud Fig. 1(b), corrugation Fig. 1(c), squat Fig. 1(e-g), and others Fig. 1(d). The labels also differentiate these defects into severity levels i.e. heavy, medium, and low. As an example, heavy, medium, and low levels of type squat are shown in Fig. 1(e), Fig. 1(f), and Fig. 1(g) respectively. All these defects along with their severity levels are used to train the ABA classifier. On the other hand, the relevant Rolling Contact Defects (RCF) i.e. squat and corrugation are further classified using image-based classification. Tab. 1 and Tab. 2 detail the distribution of available labels, with their severity levels, for the ABA and image classifier respectively.

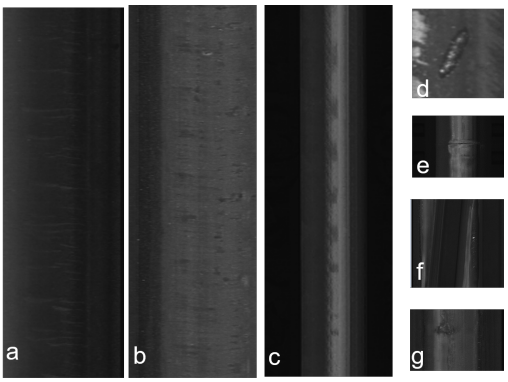

Figure 1: **Labelled defects in camera data.**

The severity level heavy is underrepresented in the labeled data for

Table 1: **Distribution of all given defects with severity levels.**

Severity level	No. of labels
Heavy	50
Medium	131
Low	917
Total	1098

types squat and corrugation, see Tab. 2, which makes it impossible to have a further classification of these types into their severity levels. Consequently, the image classifier is trained to distinguish among these types without considering the severity levels.

Table 2: **Distribution of classes with severity levels available for image classifier.**

Severity level	Type	No. of labels
Heavy	Squat	18
Medium	Squat	58
Low	Squat	633
Heavy	Corrugation	0
Medium	Corrugation	5
Low	Corrugation	23
Total		737

2.1 ABA Preprocessing

ABA data strongly depend on vehicle speed [2]. To compensate for that the ABA data is scaled by v_0/v, the quotient between a reference speed v_0 (here: 60 km/h) and the actual speed of the vehicle v. In order to perform the classification of the ABA data, the original time series is divided into short time series of 1,000 samples in length. At an ABA sampling rate of 26,500 Hz, this corresponds to a length of approximately 0.38 seconds and 0.65 meters at a speed of 60 km/h. Labels are extracted from the image bounding boxes and assigned to each of these time series. Data that contain welds and joints are excluded. Separating objects from defects is an important, yet difficult task which is not in the scope of this research. The resulting time series instances and labels are listed in Tab. 3. Fig. 2 shows raw ABA data and the corresponding label extracted from the images at a severe squat. It can be seen that the ABA peaks directly after the squat.

Table 3: **No. of labeled instances used by ABA classifier per class after preprocessing.**

Severity level	No. of labels
Intact	9794
Heavy	228
Medium	765
Low	12023
Total	22810

2.2 Image Preprocessing

Corrugation, in contrast to squats, is an elongated rail surface defect. Hence, the shape of generated true labels for corrugation is elongated, too. The length of the bounding box varies from 1,300 pixels to 13,000 pixels. To generate enough labels and introduce homogeneity in the labeled data, the width-to-height ratio of all the

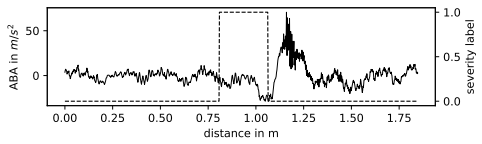

Figure 2: **ABA data (vertical component, solid line) at a heavy squat. The dashed line represents the severity label extracted from the images.**

labels is brought to 1:2 for the image classifier. The generated instances are divided into training, validation, and test dataset with a ratio of 80 %, 10 %, and 10 %, respectively, as shown in Tab. 4.

Table 4: **No. of labeled instances used by image classifier per class after preprocessing.**

Classes	Squat	Corrugation
Training set	644	952
Validation set	81	119
Test set	119	118
Total instances	805	1189

3. CNN for Image and ABA Analysis

3.1 ABA Classifier

A fully convolutional network (FCN) architecture is used for classification of the ABA time series (Fig. 3). The architecture follows the idea proposed in [20]. 1-D convolutional layers followed by batch normalization (BN) and ReLU layers build the main block of the FCN. Convolutions are carried out in three layers with the number of filters $\{16, 32, 16\}$ and filter sizes $\{8, 5, 3\}$, respectively. The convolutional layers work as feature extractor. Conventional pooling layers between the convolutional layers are omitted. Instead, a global average pooling (GAP) layer is used to reduce the feature time series produced by the convolutional layers to scalar features (1x1 feature maps). This largely reduces the number of weights compared to flattening the feature time series to a feature vector and subsequently using a fully connected layer for classification. Furthermore, the use of GAP fully preserves the shift-invariance of the convolutional operation. Batch normalization is applied to improve the convergence speed and generalization. A softmax logistic regression layer at the end produces the final class labels.

Defect detection and severity classification is difficult to perform within one network. Therefore, a step-wise classification is carried out. First, a binary classification is used to classify the data into defective and non-defective. Second, a multinomial classification is applied to the defect class that further classifies the defects by their severity level. For both classification tasks the same FCN architecture is used with the only difference in the softmax layer (Fig. 3). The input to the network is the pre-processed ABA data that consists of six channels in total (longitudinal, lateral and vertical component on the left and ride side of the axle).

3.2 Image Classifier

With the recent advances in deep learning approaches for image data analysis, selecting an appropriate model is an extensive task. There are a few application-specific criteria to consider for selecting a suitable network architecture. For example, for a detection system to work adequately not only confidence in the accuracy but also the speed of the prediction is a vital criterion. One challenge specific to this problem is that the defects appear on the rails infrequently, which introduces an imbalance in the background (normal rail) and foreground (defects) classes. If the skewed class distribution is not addressed during the training phase, it may result in the network ignoring the non-dominating class and still achieving a reduction in the training loss. The aspects like class imbalance, accuracy, and speed of inference have been considered for selecting the used deep neural network architecture. RetinaNet [12], as shown in Fig. 4, is a one-stage object detector that utilizes the focal loss function to address class imbalance during the training phase. Focal loss [12] applies a modulating term to cross-entropy loss to focus learning on the hard-negative examples as well. RetinaNet [12] also performs well on the inference speed and accuracy criteria. It consists of four major components, a bottom-up pathway, the top-down pathway and lateral connection, and two task-specific subnetworks running in parallel. A residual network, ResNet-50 [4], is deployed as a bottom-up pathway or the backbone network to extract the feature maps. Feature Pyramid Network [11], for a top-down pathway, is used to up-sample the spatially coarse features maps. Finally, the lateral connections bring together the features from the top-down and bottom-up layers. One of the subnetworks is termed as box regression head and the other as classification head. The regression subnetwork, a feature convolution network, learns to draw the anchor boxes as close to bounding boxes (labels) by drawing several regression boxes at each location while reducing the loss function. The classification subnetwork estimates the probability of the presence of a particular class instance for each anchor box and total object classes.

The DNNs are known to have a high dependence on large amounts of labeled data, which is not available for the use case. This challenge is addressed by deploying Transfer Learning [17]. In transfer learning, the basic feature maps are learned on the relatively bigger dataset and then the task-specific, but smaller dataset is used to extract the unique features belonging to the required classification. For the bigger dataset, the COCO dataset [13] and its pre-trained weights as the initial weights for training the classifier on the defects-specific dataset are used.

4. Results and Discussion

4.1 ABA Classifier

The two FCN for ABA classification are trained separately. The first FCN is trained on all instances listed in Tab. 3. By merging all defect labels (low, medium, heavy) into one class, this is reduced to a binary classification problem. The second FCN is trained on the defects only, to learn to differentiate between defects with low, medium, and heavy severity. During inference, the second FCN is only fed with the predicted defects from the first FCN. The testing results show a good performance of the two-step ABA classification with a weighted average f1-score of 93 %. From Tab. 5 it can be seen that the classes with the most incidences (intact and low severity) have the best classification results. With only 45 incidences the defects with heavy severity could be predicted with an f1-score of 84 %. However, seven heavy severity instances were classified as intact (Fig. 5).

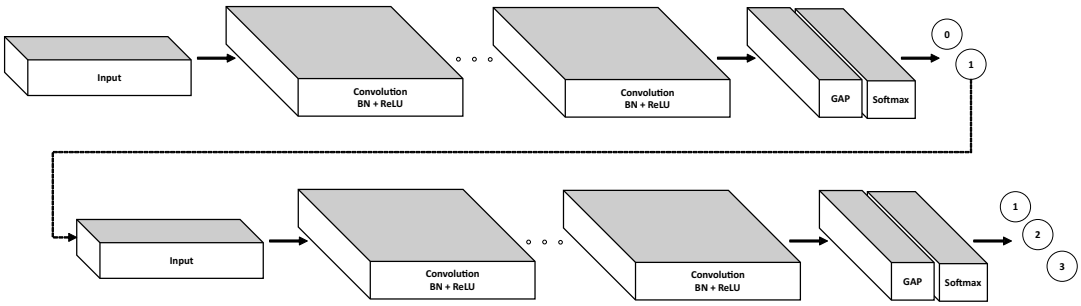

Figure 3: **Architecture of FCN.**

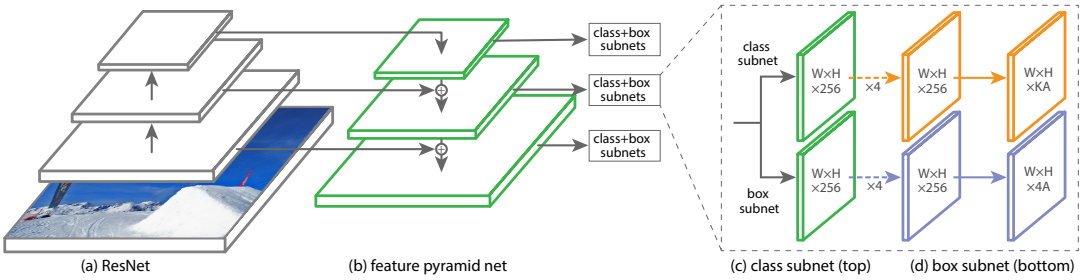

Figure 4: **Architecture of RetinaNet [12].**

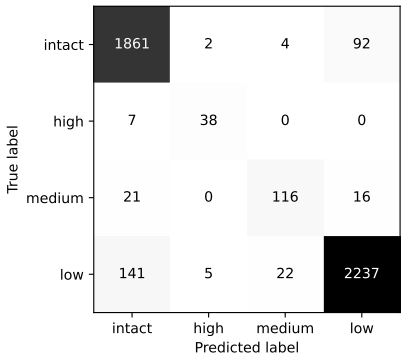

Figure 5: **Confusion matrix of ABA classification.**

Table 5: **Performance parameters for ABA classification.**

Performance Parameters	Intact	Low	Medium	Heavy
Recall (TP/TP+FN))	95 %	93 %	76 %	84 %
Precision (TP/(TP+FP))	92 %	95 %	82 %	84 %
FDR (FP/(FP+TP))	8 %	5 %	18 %	16 %
F1-Score (2TP/(2TP+FP+FN))	93 %	94 %	79 %	84 %

4.2 Image Classifier

The labeled image data, as described in Tab. 4, is used to train the proposed model, as explained in Sec. 3.2, by introducing dropout layers and batch normalization to avoid overfitting. A batch size of 4 is used, as supported by NVIDIA GPU- GeForce GTX 1080 Ti. The hyperparameters related to the loss function and learning rate, Adam [8], are learned with the help of the validation dataset.

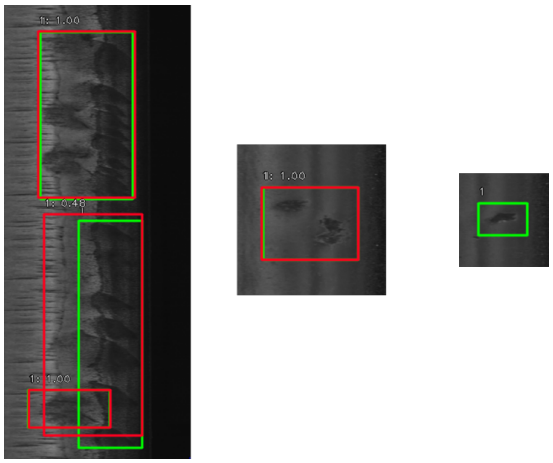

Figure 6: **Classification results for squats.**

Fig. 6 shows the detections produced on the test set by the network trained for squat detection. Red boxes represent the trained network's generated detections, with the network's confidence for each classification. Green boxes show the true labels. In Fig. 6, the left two images depict good detection performance by the network with high confidence, while the rightmost image is a squat not detected by the network. Interestingly, this network has produced additional detections (false positives) for the defect type squat, as

shown in Fig. 7 with high confidence, which were not labeled during the labeling process.
As shown in Tab. 6 as well, false negatives and false positives are quite high for the squats detection network. False positives could indicate missed labels during the manual labeling phase. However, false negatives should be reduced by enhancing the available labeled data and performing iterative training of the network.

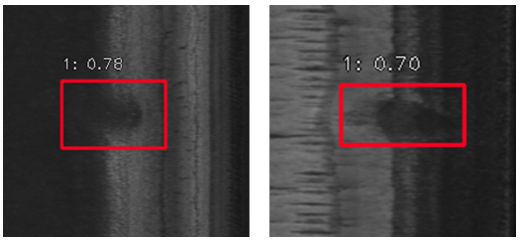

Figure 7: **False positives detected for squats.**

Fig. 8 displays the predictions of the network trained for corrugation detection. The right-side image shows the detection of all divided annotations, as explained in Sec. 2.2 belonging to one corrugation instance. On the left side, only one sub-annotation is detected out of 4. For image classification, it is a missed detection. However, as one instance of corrugation is divided into multiple labels, one such detection out of many is enough to identify the presence of corrugation on the rail from the maintenance perspective.

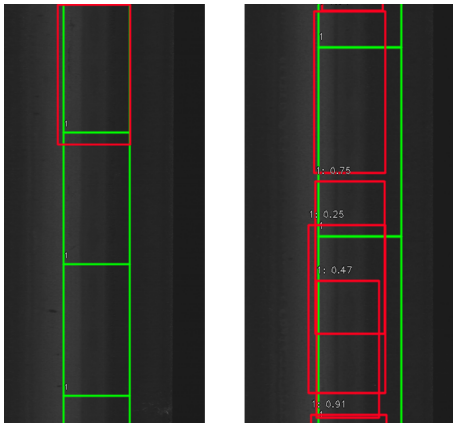

Figure 8: **Classification results for corrugation.**

With the amount of given labeled training data, parameters like recall, precision, and f1-score of the network are quite high for the squats detection, see Tab. 7. Overall, the false discovery rate, as given in Tab. 7 and Tab. 5, is an important indicator for squat and severity detection. These detections require further investigation if these are the missed instances during the manual labeling process. As seen in Fig. 7, false positives in the case of squats do appear like an actual defect. This illustrates the potential of using neural networks to produce sudo labels instead of only relying on manual labor.

Table 6: **Confusion matrix for squat detection.**

Predicted labels	True labels	
	Squat	Not squat
Squat	TP= 59	FP= 19
Not squat	FN=24	TN=-

Table 7: **Performance parameters for squat detection.**

Performance Parameters	Values (in %)
Recall (TP/TP+FN))	71.08
Precision (TP/(TP+FP))	75.64
FDR (FP/(FP+TP))	24.35
F1-Score (2TP/(2TP+FP+FN))	73.29

Tab. 8 shows the confusion matrix of the network trained for detecting corrugation. False negatives may appear high but detecting one corrugation annotation out of many, belonging to a single instance of corrugation is enough in the practical use case.
As the number of instances available for training the corrugation

Table 8: **Confusion matrix for corrugation detection.**

Predicted labels	True labels	
	Corrugation	Not corrugation
Corrugation	TP= 79	FP= -
Not corrugation	FN=39	TN=-

network, see Tab. 4, is slightly higher than squat hence parameters recall, precision and f1-score of the networks are higher as well, see Tab. 9. There are no false positives detected in the case of corrugation, unlike squat, so FDR is 0.

Table 9: **Performance parameters for corrugation detection.**

Performance Parameters	Values (in %)
Recall (TP/TP+FN))	66.95
Precision (TP/(TP+FP))	100
FDR (FP/(FP+TP))	0.0
F1-Score (2TP/(2TP+FP+FN))	80.20

As mentioned in Sec. 3.2, special attention is given to the speed of detection for real-time applications, since the collected image data is high resolution (840 x 40000 pixels per image). The inference time for the networks using a single Graphical Processing Unit (GPU) of NVIDIA - GeForce GTX 1080 is 17.5ms per image. That offers a detection speed of 57 frames per second (fps).

5. Conclusion

In this research, deep learning classifiers based on CNN are utilized on two completely different types of sensor data, namely images and ABA. Both data sets complement each other. The images are superior in discriminating different types of rail surface defects, while the dynamic vehicle-track interaction measured via ABA indicates the severity of those defects. High values of recall, precision, and f1 scores are achieved by the corresponding networks

using a limited number of labeled data. Therefore, the networks are promising for generating quality labels that can be checked by the expert. Accepted labels can be used in a second iteration of training the models. Several such training iterations are expected to increase the model performance, prediction accuracy, and reduce the manual efforts further. In this study, the networks for the different data sets were trained independently. As the next steps, a single network fusing both data sets shall be trained and evaluated.

ACKNOWLEDGMENTS

The authors thank Strukton Rail for their support on this work. This project has received funding from the Shift2Rail Joint Undertaking (JU) under grant agreement No 881574. The JU receives support from the European Union's Horizon 2020 research and innovation program and the Shift2Rail JU members other than the Union.

REFERENCES

[1] Ahmad, W. (2019). *Artificial intelligence based condition monitoring of rail infrastructure*. PhD thesis, University of Twente, The Netherlands.

[2] Baasch, B., Roth, M. H., and Groos, J. C. (2018). In-service condition monitoring of rail tracks: On an on-board low-cost multi-sensor system for condition based maintenance of railway tracks. *Internationales Verkehrswesen*, 70(1):76–79.

[3] Faghih-Roohi, S., Hajizadeh, S., Núñez, A., Babuska, R., and De Schutter, B. (2016). Deep convolutional neural networks for detection of rail surface defects. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 2584–2589.

[4] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778.

[5] Heusel, J., Baasch, B., Riedler, W., Roth, M., Shankar, S., and Groos, J. C. (2022). Detecting corrugation defects in harbour railway networks using axle-box acceleration data. *Insight - Non-Destructive Testing and Condition Monitoring*, 64(7):404–410.

[6] Hoelzl, C., Ancu, L., Grossmann, H., Ferrari, D., Dertimanis, V., and Chatzi, E. (2022). Classification of Rail Irregularities from Axle Box Accelerations Using Random Forests and Convolutional Neural Networks. In *Data Science in Engineering, Volume 9*, pages 91–97.

[7] Jang, J., Shin, M., Lim, S., Park, J., Kim, J., and Paik, J. (2019). Intelligent Image-Based Railway Inspection System Using Deep Learning-Based Object Detection and Weber Contrast-Based Image Comparison. *Sensors*, 19(21):4738.

[8] Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980.

[9] Li, H., Wang, F., Liu, J., Song, H., Hou, Z., and Dai, P. (2022). Ensemble model for rail surface defects detection. *pone*, 17(5):e0268518.

[10] Li, S., Núñez, A., Li, Z., and Dollevoet, R. (2015). Automatic Detection of Corrugation: Preliminary Results in the Dutch Network Using Axle Box Acceleration Measurements. In *2015 Joint Rail Conference*, page 7.

[11] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature Pyramid Networks for Object Detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944.

[12] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2018). Focal Loss for Dense Object Detection. *Computer Vision and Pattern Recognition (CVPR)*, pages 318–326.

[13] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, volume 8693, pages 740–755. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

[14] Loidolt, M. and Marschnig, S. (2022). Evaluating Short-Wave Effects in Railway Track Using the Rail Surface Signal. *Applied Sciences*, 12(5):2529.

[15] Min, Y., Xiao, B., Dang, J., Yue, B., and Cheng, T. (2018). Real time detection system for rail surface defects based on machine vision. *EURASIP Journal on Image and Video Processing*, 2018(1):3.

[16] Molodova, M., Li, Z., Nunez, A., and Dollevoet, R. (2014). Automatic Detection of Squats in Railway Infrastructure. *IEEE Transactions on Intelligent Transportation Systems*, 15(5):1980–1990.

[17] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. (2018). A Survey on Deep Transfer Learning. *The 27th International Conference on Artificial Neural Networks (ICANN 2018)*.

[18] Tastimur, C., Karakose, M., Akın, E., and Aydın, I. (2016). Rail defect detection with real time image processing technique. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, pages 411–415.

[19] Toliyat, H., Abbaszadeh, K., Rahimian, M., and Olson, L. (2003). Rail defect diagnosis using wavelet packet decomposition. *IEEE Transactions on Industry Applications*, 39(5):1454–1461.

[20] Wang, Z., Yan, W., and Oates, T. (2016). Time series classification from scratch with deep neural networks: A strong baseline. *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1578–1585.

[21] Westeon, P. F., Ling, C. S., Roberts, C., Goodman, C. J., Li, P., and Goodall, R. M. (2007). Monitoring vertical track irregularity from in-service railway vehicles. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 221(1):75–88.