

Review

# A Review of Deep Learning Applications for Railway Safety

Kyutaek Oh <sup>1</sup>, Mintaek Yoo <sup>2,\*</sup>, Nayoung Jin <sup>3</sup>, Jisu Ko <sup>3</sup>, Jeonguk Seo <sup>3</sup>, Hyojin Joo <sup>1</sup> and Minsam Ko <sup>1,\*</sup><sup>1</sup> Department of Human-Computer Interaction, Hanyang University, Ansan-si 15588, Korea<sup>2</sup> Railroad Structure Research Team, Korea Railroad Research Institute (KRRRI), Uiwang-si 16105, Korea<sup>3</sup> Department of Applied Artificial Intelligence, Hanyang University, Ansan-si 15588, Korea

\* Correspondence: thezes03@krrri.re.kr (M.Y.); minsam@hanyang.ac.kr (M.K.)

**Abstract:** Railways speedily transport many people and goods nationwide, so railway accidents can pose immense damage. However, the infrastructure of railways is so complex that its maintenance is challenging and expensive. Therefore, using artificial intelligence for railway safety has attracted many researchers. This paper examines artificial intelligence applications for railway safety, mainly focusing on deep learning approaches. This paper first introduces deep learning methods widely used for railway safety. Then, we investigated and classified earlier studies into four representative application areas: (1) railway infrastructure (catenary, surface, components, and geometry), (2) train body and bogie (door, wheel, suspension, bearing, etc.), (3) operation (railway detection, railroad trespassing, wind risk, train running safety, etc.), and (4) station (air quality control, accident prevention, etc.). We present fundamental problems and popular approaches for each application area. Finally, based on the literature reviews, we discuss the opportunities and challenges of artificial intelligence for railway safety.

**Keywords:** railway; railway safety; deep learning; AI application



**Citation:** Oh, K.; Yoo, M.; Jin, N.; Ko, J.; Seo, J.; Joo, H.; Ko, M. A Review of Deep Learning Applications for Railway Safety. *Appl. Sci.* **2022**, *12*, 10572. <https://doi.org/10.3390/app122010572>

Academic Editor: Sangtae Ahn

Received: 1 September 2022

Accepted: 4 October 2022

Published: 19 October 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Artificial intelligence, which began in the 1950s with the question “Can computers think?”, has become a modern concept that means “a computing-based technology system that automates intelligent tasks generally performed by ordinary people.” [1–3]. Artificial intelligence has been widely used in various fields and has advanced sensing and data processing technologies. This paper presents an overview of deep learning applications for railway safety. We analyzed earlier studies over four representative application areas: (1) railway infrastructure, (2) train body and bogie, (3) operation (train running), and (4) station.

There have been many studies monitoring railway infrastructure, such as catenary and rail surfaces. Railway defects can occur for diverse reasons, such as long-term accumulated operation, rain, sunlight, wind, etc. Regular inspections are essential since rail defects can cause significant accidents. Recently, many studies have been conducted to detect defects in railways and related parts with rapidly developed artificial intelligence technology to prevent railway accidents. Diverse data sources have been used for detecting railway defects, such as images [4,5], accelerometers [6,7], and ultrasonic sensors [8,9].

There have also been many studies conducted on detecting train defects using artificial intelligence. The railway train has a complex structure that combines various parts, such as vehicle wheels, split pins, tram lines, and pantographs. Each train accessory has the characteristic that the degree of corrosion or decrease in durability is not constant because of the difference in function and environment. Fault detection and prediction are essential because even small-area or early-progress defects (cracks, cuts, aging, etc.) on trains can cause severe threats to passenger safety. These features can be even more critical for high-speed trains as their parts are exposed to harsh environments compared to other trains.

Additionally, safety accidents during train operation can occur, such as railroad trespassing or derailment. Such train accidents during train operation can lead to many casualties and various artificial intelligence technologies for train operation safety have been studied. For example, segmentation methods based on deep learning models have been proposed to detect rail tracks during operation or to check the existence of any obstacles on railways (e.g., people, cars). In addition, there have been attempts to quantify wind risk or train running safety for controlling and managing further operation.

Finally, many studies have been conducted wherein artificial intelligence is used for railway station safety. A railway station is dynamic and complex due to the presence of many people, including passengers and station staff, and trains that stop and depart quickly. Therefore, it is necessary to prevent and deal with various safety accidents that may occur in stations. For example, diverse artificial intelligence models were developed to quickly identify three different types of safety incidents (fall, slip, and trip) [10] and to monitor air quality in a station. Furthermore, many studies have modeled the station as a dynamic and complex system.

This paper provides a comprehensive view of railway safety by covering the four representative areas. Table 1 shows the railway safety areas addressed by earlier reviews. While most review papers have focused on a specific problem, few studies covered various areas. For example, Tang et al. [11] and Liu et al. [12] covered the four application areas we covered. However, these studies aimed to overview artificial intelligence for the railway, so some specific safety issues were rarely covered. This review more widely includes studies about railway safety. For example, it further covers safety issues related to the catenary in the railway infra, the train's door and suspension, and wind risk during the operation. In addition, although many studies have focused on visual inspection methods based on image data, this study includes diverse data types. For example, various sensor data (e.g., vibration, current, acoustic emission signals) and image-shaped data (e.g., 2D-camera images and laser ultrasound scanning data) are explained in this review.

**Table 1.** Coverage of Existing Reviews about the Railway Safety.

Papers		Application Areas <sup>1</sup>				Data Types	
Ref	Year	Railway Infra	Train	Operation	Station	Image	Others <sup>2</sup>
Tang et al. [11]	2022	○	△	△	△	○	○
Liu et al. [12]	2019	○	○	△	△	○	×
Ghofrani et al. [13]	2018	△	△	△	×	○	○
Hu et al. [14]	2021	△	×	×	×	○	×
Sedghi et al. [15]	2021	△	△	△	△	×	△
Yin et al. [16]	2020	△	×	×	△	○	△
Wen et al. [17]	2019	×	×	×	△	○	○
Chenariyan et al. [18]	2019	△	×	×	×	○	△
This study	2022	○	○	○	○	○	○

<sup>1</sup> ○: almost subdomains listed in Sections 4–7, △: about half of the subdomains, ×: rarely covered. <sup>2</sup> ○: more than two other types, △: one type, ×: none.

The subsequent parts of this review are organized as follows: Section 2 provides an overview of deep learning approaches that have been used for railway safety. We classified the deep learning methods according to their data source and task. In Section 3, we described the methodology for searching and analyzing related studies. From Section 4 to Section 7, we explained the four application domains in railway safety (i.e., railway infra, train, operation, and station) and representative studies. Lastly, Section 8 concludes the paper by discussing the opportunities and challenges of artificial intelligence for railway safety.

## 2. Overview of Deep Learning Approaches

A deep neural network (DNN) is a machine learning that emulates brain neuron cells. Therefore, DNN can extract patterns and features like the human brain from numerous datasets. DNN is constructed of several layers (e.g., input layer, hidden layer, output layer). The input layer is responsible for receiving input values. On the other hand, the output layer is responsible for output values. There can be many hidden layers between the output and input layers. The greater the number of these hidden layers, the deeper the neural network. In the hidden layer, the following operations are performed:

$$\mathbf{u} = \mathbf{W} \times \mathbf{x} + \mathbf{b} \quad (1)$$

$$\mathbf{z} = f(\mathbf{u}) \quad (2)$$

where  $\mathbf{x}$  is input vector.  $\mathbf{W}$  and  $\mathbf{b}$  are the weight matrix and the bias term, respectively, and these are updated through training.  $f(\mathbf{u})$  is the activation function that makes neural networks nonlinear.  $\mathbf{z}$  is the output vector of the hidden layer.

Diverse deep learning architectures have been studied by extending the DNN structure. They can be distinguished according to their data types and tasks.

### 2.1. Data Types

#### 2.1.1. Image Data

Image data are represented in a two-dimensional structure in the form of a numerical matrix consisting of points called pixels. Pixels represent the contrast of colors in numbers ranging from 0 to 255, where 0 is black and 255 is white. In the case of a color image, the color of the image is expressed using the light and shade of red, green, and blue with three channels of RGB. Before processing images using a deep learning model, image preprocessing, such as image alignment, cropping, and adjustment (e.g., brightness and contrast), can be conducted.

A convolutional neural network (CNN) is a representative deep neural network used in image data processing. CNN can extract image patterns with spatial structures because filters composed of multiple weight values move spatially. The following equation represents the convolution in CNN:

$$\mathbf{U}_{i,j} = (\mathbf{W} \times \mathbf{X})_{i,j} = \sum_m \sum_n \mathbf{X}_{m,n} \mathbf{W}_{i-m,j-n} \quad (3)$$

$$\mathbf{Z} = f(\mathbf{U}) \quad (4)$$

where  $i, j$  are the index of the output matrix, and  $m, n$  are the index of the input matrix.  $*$  is convolution.  $\mathbf{X}$  is the input matrix.  $\mathbf{X}_{m,n}$  is the value of row  $m$  and column  $n$  of  $\mathbf{X}$ .  $\mathbf{Z}$  is the output matrix of the convolution layer.

A CNN performs convolutional operations commonly used in image or signal processing. The CNN moves a mask, also called a kernel, filter, or window, and performs convolutional operations with input data to extract data features. Because this approach allows the detection of relevance between one element and neighboring elements, CNNs are suitable for data processing with grid structures, such as images. CNNs have shown superior performance to humans in some complex image processing problems and have also contributed significantly to image retrieval services, autonomous vehicles, and image automatic classification systems.

#### 2.1.2. Time-Series Data

Sequential data refer to data in which objects in the data set have a certain order. Sequential data include numerous kinds of time series data with temporary properties, such as language, stock quotes, electrocardiogram (ECG) signals, seismic waves, and DNA sequencing. In railway safety, sensor data (e.g., vibration) and video (e.g., CCTV) are prevalent time-series data types.

The recurrent neural network (RNN) was developed to deliver information that occurred at the previous time step to the next time step through the recurrent edge, which is the edge connecting the hidden nodes. In other words, an RNN hidden layer can remember important things about input information that allows them to predict what will come next. The key operation in RNN can be described by the following formula:

$$\mathbf{a}^{(t)} = \mathbf{b} + \mathbf{W}\mathbf{h}^{(t-1)} + \mathbf{U}\mathbf{x}^{(t)} \quad (5)$$

$$\mathbf{h}^{(t)} = \tanh(\mathbf{a}^{(t)}) \quad (6)$$

$$\mathbf{o}^{(t)} = \mathbf{c} + \mathbf{V}\mathbf{h}^{(t)} \quad (7)$$

where  $\mathbf{U}, \mathbf{V}, \mathbf{W}$  is a weight matrix that is updated through training, for input-to-hidden, hidden-to-output and hidden-to-hidden.  $\mathbf{b}, \mathbf{c}$  are bias vectors.  $\mathbf{x}^{(t)}$  is input vector.  $t$  is the time step.  $\mathbf{o}^{(t)}$  is output vector.

Long short term memory (LSTM) improves an original RNN structure by adding gates that select inputs and outputs at a time step to properly understand the contextual dependence of sequential data (e.g., long-term dependence). A gated recurrent unit (GRU) is also a variant of RNN, like LSTM, but has fewer parameters.

## 2.2. Tasks

### 2.2.1. Classification

Classification is a sort of supervised learning that is the process of identifying the category relationship of existing data and determining the category of newly observed data by itself. In the field of images, it is used to assign an appropriate label (or class) to objects in a given image as input. For example, a classification model can be trained to recognize a number in handwritten images.

There are various types of image classification models. Visual geometry group (VGG) is a relatively early classification model developed to determine how the depth (number of layers) of neural networks affect performance [19]. VGG has a structure that combines convolutional layers for feature representation and fully connected layers for classification. Filters of  $3 \times 3$  are used to reduce the number of model weights that require learning to efficiently increase the depth of the model.

Residual Net (ResNet) [20] is a deep learning network with 152 layers. For ResNet, a new concept called residual block was introduced. Unlike previous networks, which aimed to generate output values as similar as possible to the correct answer, ResNet was designed to minimize the residual (the difference between the output and input values). This approach makes it possible to preserve previously learned information and to consider only additionally learned information. DenseNet [21] is similar to ResNet, but it uses the operation of the concatenation of the output of the previous layer with the next layer.

Res2Net [22] is a structure that combines ResNet with DenseNet and is a classification model that leads to performance improvement by configuring hierarchical residual-like connections in a bottleneck residual block. Res2Net is also characterized by segmentation by increasing the range of receptive fields in each network layer rather than expressing multi-scale layer-wise features.

Finally, Inception is a neural network structure designed to address problems that arise when classification models of deep and wide structures learn [23]. The number of channels was reduced while maintaining the input form using a convolutional layer of a  $1 \times 1$  filter and matrix operations were densely performed to increase the computational efficiency. In addition, Inception uses an auxiliary layer that delivers backpropagation by calculating the intermediate learning error, to convey information to the deep layer during learning, and batch normalization, to prevent overfitting that frequently occurs in deep learning.

### 2.2.2. Object Detection

Object detection and localization have been popular tasks for railway safety. Object detection refers to a task that performs both classification and localization on multiple objects. Localization is a task to display the location of a specific object in an image through a bounding box. Object detection methods can be categorized into single-stage and two-stage methods. A single-stage method detects the potential locations of the target object and classifies them by a single network. The two-stage method separately performs a region proposal first, extracts possible areas that the target object can locate, and then selects and classifies regions.

Regions with CNN features (R-CNN) is a two-stage method that performs the task of proposing an object region and classifying objects separately. The R-CNN first extracts regions by the selective search algorithm and then uses a pre-trained CNN model to extract image features for the classifier to distinguish the object and regressor to localize the object. In R-CNN, learning and inference are slow during region proposal and image feature extraction from regions because it processes about 2000 regions per image. Faster R-CNN solves the bottleneck phenomenon that occurs when proposing regions [24]. It first extracts image features from an input image and then performs the region proposal. Faster R-CNN uses a region proposal network, a deep learning network runnable on a GPU, to improve the region proposal extraction process by selective search.

You Only Look Once (YOLO) is a single-stage method for object detection using a single neural network to perform both classifying and detecting of the potential location area of the target objects [25]. In YOLO, the convolutional layers extract feature maps and fully connected layers and then predict the bounding box and class probability. YOLO divides the input image into  $S \times S$  grids, and bounding box coordinates and confidence scores are predicted for each grid. Since the whole detection pipeline is a single network, YOLO can be optimized end-to-end directly on detection performance. Single-shot detectors (SSD) [26] is also a single-stage method for object detection. This method begins with the idea that a single feature map may be insufficient to detect objects of various sizes. SSD predicts bounding boxes using a pyramidal feature hierarchy instead of image grids in YOLO. The pyramidal feature hierarchy consists of feature maps extracted from various layers using a single deep neural network. Each convolutional layer has a different receptive field size and can provide unique image features at different scales.

In the object detection problem, the number of objects in the image is generally small, so it is easy to develop a class balance problem with very few object areas compared to the background area. RetinaNet [27] is a model that applies focal loss designed to focus on hard negative samples by lowering the weights for easy samples. In addition, both local and global features are utilized by adding a spatial attention map block (SAMB) and a channel weight map block (CWMB) in the image feature extraction process. This allows RetinaNet to weaken the influence of the background in the object detection process and focus on important features.

### 2.2.3. Segmentation

Segmentation is a method of extracting an object of interest from an image in units of pixels. By giving each pixel a label, it is possible to know which pixel belongs to which object. Segmentation is necessary for identifying shapes of target objects in detail, such as in traffic safety, autonomous driving services, and in reading magnetic resonance imaging (MRI). Depending on the purpose of use, segmentation can be divided into semantic segmentation and instance segmentation. Semantic segmentation assigns a class label to every pixel in an image, such as a person or car. The objects of the same class have the same label. However, instance segmentation identifies each object separately, even if they belong to the same class.

You Only Look At CoefficientS (YOLACT) is a real-time model that improves the processing speed of instance segmentation by omitting the localization step [28]. Instead, this model solves the problem by dividing the segmentation process into two parallel,

instead of sequential, subtasks. The first task generates a dictionary of non-local prototype masks over the entire image and another predicts the linear combination coefficient for each instance. Then, YOLACT produces instance masks by linearly combining the prototypes with the mask coefficients.

#### 2.2.4. Feature Extraction

Feature extraction transforms raw data into numerical features more beneficial for the main task (e.g., classification, object detection). This task often affects model performance, helping reduce the dimensionality of the model and better represent latent patterns. Principle component analysis (PCA) is one of the traditional algorithms for feature extraction. PCA is a method of reducing multidimensional data by selecting the axis with the largest variance as the first principal component, then selecting the larger axis as the second principal component, and linearly converting the data when each variable (feature) is projected onto one axis. Other known techniques include linear discriminant analysis (LDA) [29], canonical correlation analysis (CCA) [30], singular value decomposition [31], isometric feature mapping (ISOMAP) [32], and locally linear embedding (LLE) [33].

Auto encoder (AE) is a deep neural network that can be used for feature extraction [34]. AE is used for anomaly detection, which determines whether a sample is normal or abnormal, or for denoising operations that extract the original data by removing the noise added to the data. AE is unsupervised learning that learns to output the same results as the input data. However, since the dimension of the hidden layer is designed to be lower than that of the input and output layers, AE learns in the direction of exploring representation information that can effectively indicate input data. The restricted Boltzmann machine (RBM) is also a deep learning model for feature learning that works through the process of finding better representations of input values [35]. The RBM consists of a visible layer, which is an input layer, and a hidden layer in which feature values are learned. The deep belief network (DBN) is a probabilistic generative model built by layers of pre-trained RBMs [36].

### 3. Methodology

In this review, we aim to (1) identify problems by railway safety category and solutions using deep learning models, (2) evaluate the performance of the proposed deep learning model and comparison with the previous model, and (3) summarize supplementary points of the proposed method and additional issues to be dealt with afterward. For doing this, we searched papers including the following keywords on Google Scholar: “railway” OR “deep learning” OR “defect” OR “railroad” OR “safety” OR “artificial intelligence”. In order to investigate in-depth safety issues for each category, category-specific keywords were considered (e.g., “catenary” OR “surface”). In addition, the entire paper cited by the key reference paper was examined whether to be included in the review.

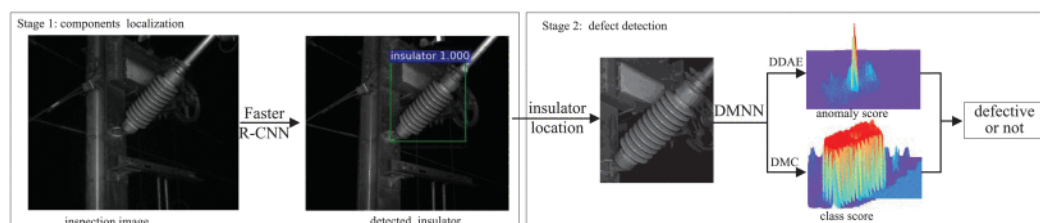
We checked the abstracts of all selected papers and excluded papers that were not related to railway safety or did not address the applications of deep learning techniques for solving problems. If it is not clear to determine about the papers, the introduction section and methodology section were additionally reexamined. Cross-checking was performed three times by independent authors. Four of the authors checked each part of the review paper. Next, two authors individually examined the whole part of this paper without discussion. If two papers had overlapping parts of the contents, a paper with a high number of citations was selected. When the data source was not clearly marked on the paper, it was classified as Custom. The performance of the model was selected as metrics and values with the best results. The papers included in the review were finally updated on September 2022. The details of the reviewed papers and the performance metrics were described in Appendix A, Tables A1 and A2, respectively.

## 4. Railway Infra Safety

### 4.1. Catenary

The catenary, which is responsible for supplying electricity to trains, is a critical facility in the electric railway system. Therefore, defects in a catenary can pose a severe threat to railway safety. While a human inspector usually needs to shut down the train power and go up to the vehicle to examine the state of the catenary, this procedure can cause many safety accidents. Prior studies have made efforts to study computer vision technologies to detect catenary defects fast and early.

Kang et al. [37] focused on detecting defects in the insulator, which is a catenary component. Figure 1 shows the proposed workflow of catenary defect detection. Their proposed framework captured images of areas where insulators are usually located using fixed-viewed cameras. Next, a Faster R-CNN model localized the specific location of the insulator in the input image (i.e., object detection). Finally, two other deep learning models were implemented to examine the extracted images of the insulator. One model was a deep learning classifier that had a CNN-DNN structure to output the classification score of the input image. The other model was an auto-encoder model that outputs the abnormal score of the insulator. The abnormal score determines whether and how the insulator is damaged. Actual data from Hefei–Fuzhou high-speed railway line was used for evaluation. The results showed that the proposed framework effectively mitigates the small data problem and the complexity of processing catenary images, which can cause a decrease in diagnosis performance.



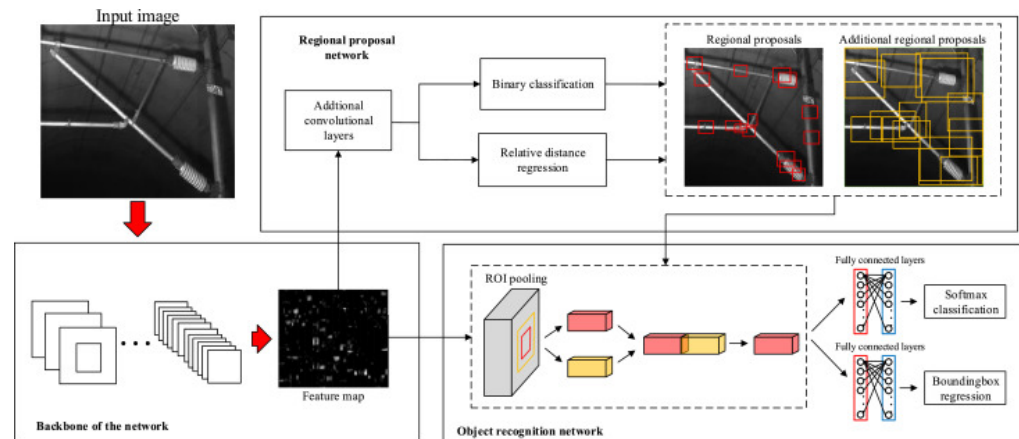
**Figure 1.** Workflow of catenary defect detection. Reproduced with permission from IEEE [37].

There have been many studies on defect detection in a dropper, which connects a catenary and a messenger wire. Guo et al. [38] proposed a method to detect defects in a dropper from image data by deep learning models based on Faster R-CNN and fully connected layers. A balanced attention feature pyramid network (BA-FPN) was proposed that integrates multiple-level features onto the original Faster R-CNN structure. This enhances detection performance by extracting useful image features from small areas from the entire catenary image where the dropper is placed. Experimental results on the VOC 2012 and MSCOCO 2014 datasets showed that the proposed models achieved higher performance than conventional detection models (86.8% at mAP@0.5 and 83.9% at mAP@0.7).

The clevis is another catenary component that is located between the registration arms and the cantilever. The Faster R-CNN has also been widely used to detect clevis defects. Han et al. [39] proposed a deep learning model that focuses on image features from the surrounding areas of the clevis, as shown in Figure 2. This idea is under the heuristic insight that the catenary has a typical structure, so there are specific areas where useful image features for clevis crack diagnosis are likely placed. The evaluation results reveal that the proposed model has higher crack detection performance than existing models, such as Faster R-CNN and YOLO. In addition, the proposed model was robust to different size, texture, and grayscale transformations that resulted from changes in shooting distance, angle, and illuminance.

The split pin combines and supports diverse components in the catenary. Wang et al. [40] studied a deep learning framework that determines three states of the split pin (missing, loosening, and normal) according to the location of the joint. First, the proposed framework

performed an object detection task based on YOLO v3 to explore split pins for five joints extracted from the entire catenary image. Next, semantic segmentation was performed in three parts (head, body, and tail) using DeepLab V3+ [41–43]. Finally, the classification model determined the state of the split pins. The evaluation was conducted with 2670 catenary images, including 21,472 split pins, and the split pin defects were detected with very high accuracy (98.72%).



**Figure 2.** Catenary Inspection Using the Faster R-CNN network. Reproduced with permission from Elsevier [39].

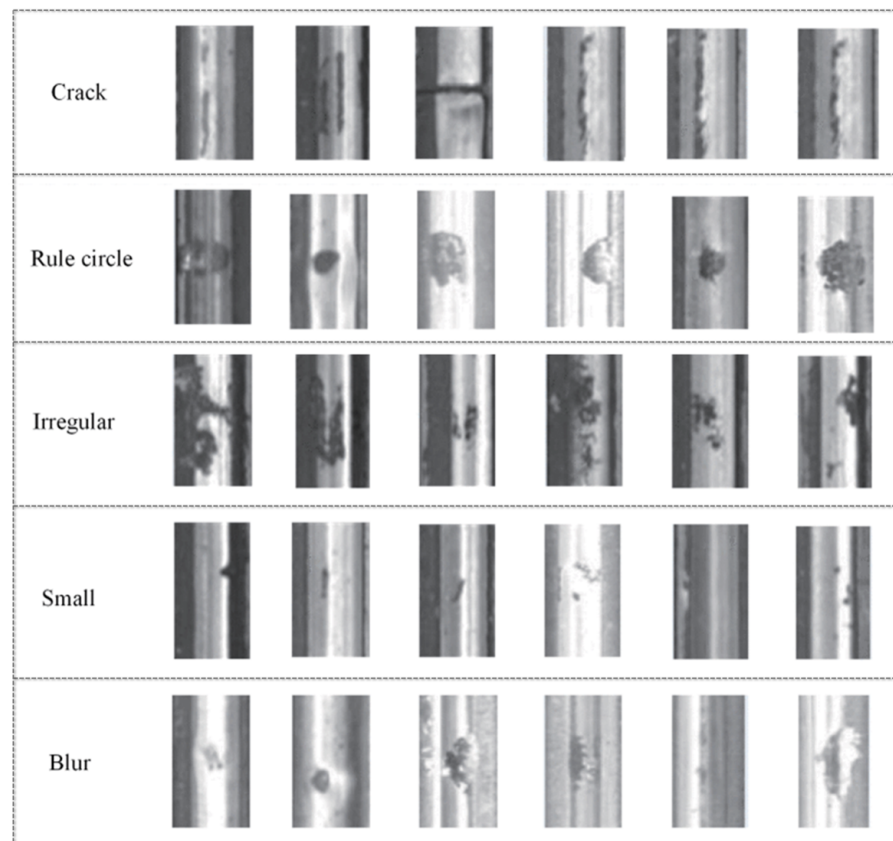
Chen et al. [44] studied an image-based deep learning model to check damage in the current-carrying ring of a catenary. RetinaNet [27] was used to detect and classify defects for fault diagnosis. RetinaNet was trained based on the focal loss that mitigates the imbalance between classes of training data, instead of cross-entropy loss. Additionally, RetinaNet contains a spatial attention map (SAM) and a channel weight map (CWM) to harness the spatial characteristics of each feature map and consider patterns in the channel. Performance tests were conducted with catenary images taken at various locations, and the proposed model achieved the best performance in diagnosis accuracy.

#### 4.2. Rail Surface

Scouring, breaking, and deficient fastening in bolts and sleepers are typical defects on the rail surface. Figure 3 presents several types of rail surface defects. Santur et al. [45] proposed a machine learning model based on image features of defects extracted based on PCA, kernel principal component analysis (KPCA), singular value decomposition (SVD), and histogram match (HM). Faghih-Roohi et al. [46] adopted deep convolution neural networks to determine defect types of surface images (normal, weld, L-squat, M-squat, S-squat, and joint). They designed and compared three CNN models (small, medium, and large), each with different structures (number of layers, number of filters, sizes of filters, activation functions). The large model outperforms small and medium models and shows about 93% accuracy in detecting surface defects.

Many studies have been performed to develop object detection methods on rail surface images. For example, Yanan et al. [47] developed a fast and accurate defect detection model for rail surfaces using YOLO v3, which has the strength of accurately and quickly detecting small-sized targets.

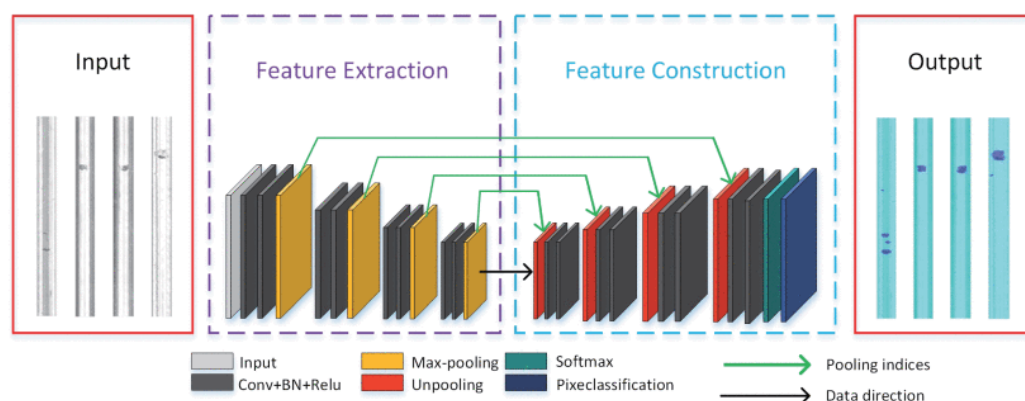




**Figure 3.** Several types of rail surface defect images. Reproduced with permission from IEEE [47].

The detection model receives  $416 \times 416$  images and divides them into boxes of various sizes, calculates normalized coordinate values of defects depending on the location of defects located inside the box, predicts defect inclusion scores for each box, and evaluates reliability. This method achieved high detection rates (97%) in 0.15 s. Similarly, Yuan et al. [48] developed a model that detects the location of defects from existing rail surface images. Their proposed model consisted of a MobileNetV2 for extracting image characteristics and a YOLOv3 module for defect localization. Their performance test results confirmed that the model increased the mean average precision (MAP) by more than 4%. Shang et al. [49] presented a novel pipeline consisting of two stages. In the first stage, an input image is localized to extract rail areas. The second stage detects defect areas using a deep learning model, a fine-tuned Inception3.

Some studies proposed deep learning methods to extract defects more detailedly using image segmentation. Kim et al. [5] adopted image segmentation to distinguish specific areas of defects on rail surfaces. The defective part was labeled in units of image pixels to train the segmentation model. The proposed model was implemented based on the VGG-19 structure and showed IoU and F1 scores exceeding 90%. Liang [50] proposed SegNet, a deep convolution neural network, to detect defects on rail surfaces. As shown in Figure 4, SegNet comprises feature extraction (FE) and feature construction (FC). This structure can learn rail surface types and their distributions from a given training dataset.



**Figure 4.** SegNet architecture for rail defect analysis. Reproduced with permission from IEEE [50].

Jiang et al. [8] proposed a technique for detecting rolling-contact fatigue (RCF), which is a failure or material removal driven by crack propagation caused by a near-surface alternating stress field. Specifically, this study used laser ultrasound scanning data to detect RCFs. To extract features from ultrasonic signals, wavelet packet transform (WPT), which decomposes signals in different frequency bands, and KPCA, which reduces the correlation between all defective features, were used. A support vector machine (SVM) model performed the final detection based on the features. A squat is an RCF defect and often leads to rail breaks. Yuan et al. [51] proposed an algorithm to automatically detect the position of rail squats using vehicle axle box acceleration signals. The convolutional variable auto encoder (CVAE), an unsupervised manager, extracts critical features from signals, and the one-class SVM (OCSVM) detects rail squats in abnormal conditions. In their study, the proposed method was shown to be robust to signal noise and train speed variability.

Suwansin and Phasukkit [9] analyzed acoustic emission signals from fatigue cracks on rails and developed a non-destructive localization model that determines the presence and location of defects without damaging railways. A DNN structure consisting of three hidden layers used the hyperbolic tangent function for considering the transient nature of acoustic emission signals. The model processed the acoustic emission signals and classified them into breaks at the head, web, or foot of the steel rail.

Shebani and Iwnicki [52] developed a neural network model that predicts wheel and rail wear using an artificial neural network. Nonlinear autoregressive models with an exogenous input neural network (NARXNN) were developed for wheel and rail wear prediction. Wheel and rail profiles, plus load, speed, yaw angle, and first and second derivative of the wheel and rail profiles, were used as inputs to the neural network while the neural network output was wheel and rail wear. Their laboratory tests confirmed the feasibility of the proposed wear prediction methods for realistic wheel and rail profiles and materials.

Studies have also been conducted to facilitate the acquisition and utilization of rail surface data necessary for artificial intelligence models. Wu et al. [53] attempted to develop a robust detection framework for the quality and sampling rates of rail surface images. Unmanned aerial vehicles (UAV), capable of moving at speeds ranging 2–15 m/s, were used to collect rail images. In addition, the proposed model used enhanced residual blocks for time and memory optimization in defect detection. Two image datasets from high-speed train sections between Beijing and Shanghai and Class I freight lines in South Carolina were used for training and testing the model.

Zhang et al. [54] proposed an efficient learning method based on line-level labels. Use of line-level labels can decrease the time and effort needed to collect data compared to pixel-level labels. In addition, this method can lower the model complexity and is more suitable for small data. The proposed model converted color information into numeric vectors using a 1D-CNN and LSTM, and detected rail surface defects line by line. Hajizadeh et al. [55] focused on the data imbalance in detecting rail surface defects. Most rail

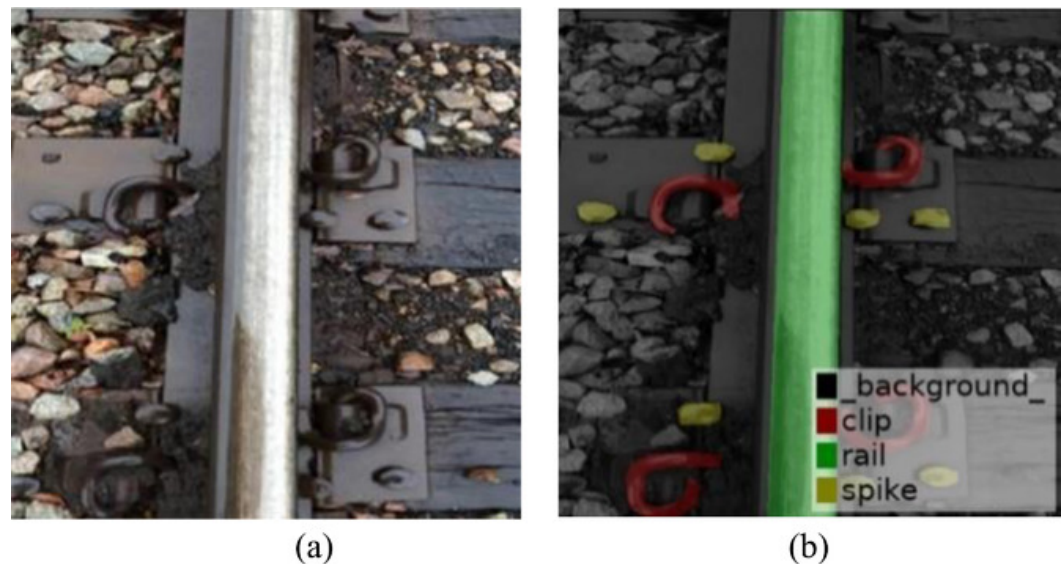
image datasets have an overwhelming proportion of normal state data than abnormal data including defects. Many captured images are not labeled to indicate whether they contain defects or not. Hajizadeh et al. [55] proposed semi-supervised learning methods to detect defects on rail surfaces. The proposed semi-supervised learning methods showed compliance performance, more than other methods, to data imbalance, such as undersampling and oversampling.

Santur et al. [56] addressed degraded image quality due to substances, such as dust or oil, which often cause false-positive cases. A high-resolution camera can also help deal with substances but leads to loss of time and additional costs in the railway maintenance process. Santur et al. [56] presented hardware and software architectures to perform railway surface inspection using a three-dimensional (3D) laser camera and deep learning. The use of 3D laser cameras in the railway inspection process provided high accuracy rates in real-time.

Falamarzi et al. [57] utilized train acceleration data to estimate the degradation of tram rails. Machine learning algorithms (Random Forest, SVM, and ANN) were trained and tested using Melbourne tram network data. The study results revealed that the proposed method allows for cost-effective maintenance strategies by reducing the time and effort in collecting data for evaluation.

#### 4.3. Rail Components

Defect inspection of rail components (e.g., spikes that secure rails to ties and clips that press down on the bottom of the rail to concrete ties) commonly depends on the judgement of individual human inspectors. Many studies have used deep learning models to improve manual rail component inspection. Guo et al. [58] proposed a framework that can detect pixel-wise rail accessories in real time using CNN-based models that receive high-resolution rail images, shown in Figure 5. Their proposed framework shows a speed of over 30 FPS in high-resolution processing video in real-time. These results show that inspection video can be quickly converted into helpful information to aid rail maintenance. Similarly, Gibert et al. [59] proposed CNN models to perform defect detection in rail ties and fixtures.



**Figure 5.** Example of original image and label result: (a) ground truth; (b) instance label visualization. Reproduced with Permission from Blackwell [58].

Sresakoolchai and Kaewunruen [60] developed a model that detects defects in rail dipped joints and track settlements and quantifies the degree of defects. Their proposed deep learning method receives 14 features, including weight, speed, and peak acceleration sensor data measured on wheels. The CNN and RNN modules in the model used time

series acceleration values, and the DNN modules used train weight, speed, and wheel acceleration feature point values.

A train delivers high acceleration to wheelsets, axle boxes, the bogie, and total vehicle bodies as it passes through the rail. If defects occur in rail components, the acceleration data show different patterns. Yang et al. [7] proposed a deep learning-based approach for defect detection in rail joints through CNNs on acceleration sensor data. CNN-based models can work directly with raw data to reduce the heavy preprocessing of feature engineering and directly detect joints located on either the left or the right rail. Similarly, Sun et al. [61] used acceleration data to detect defects on rail joints. A single CNN model was designed to detect both left and right joints together. This can mitigate the interference issue when a different model is used for each side, which increases a high false-positive rate.

A clamp is a rail component that ties a rail so it does not move from side to side. The clamp should maintain railway safety by maintaining the spaces on the left and right sides of the rail. Inspecting clamps is time-consuming and expensive because it depends on visual inspections made by a human expert. Chandran et al. [62] attempted to check clamps using two differential eddy current signals. The current signals were collected using sensors capable of measuring eddy-phase current signals of 18 kHz and 27 kHz and missing clamps in the fastening system were detected using machine learning algorithms.

Soares et al. [63] derived malfunction patterns of a rail switch machine. Mean, intermediate, maximum, and minimum values were extracted from current signals during the switch operation. Then, similar defects were formed into one group by using k-means clustering. The proposed model evaluated the performance by receiving current data generated during switch operation provided by the railway company and showed a high score (.860) in the silhouette score, a clustering performance index.

Guo et al. [64] designed a real-time monitoring system to detect rail slab deformation of high-speed railways. This work combined fiber optic sensing methods and machine learning models to identify track slab deformation by using on-site track-side vibration acceleration data. The proposed method could identify the track slab deformation effectively and the detection rate could reach 96.09%.

#### 4.4. Rail Geometry

Recent studies have utilized deep learning to analyze vibration data to evaluate railway track quality. Ma et al. [65] proposed a method to evaluate the quality of the rail track based on vehicle-body vibration. CNN and LSTM structures were integrated to process vehicle-body accelerations and predict vertical vehicle-body vibration. Such vehicle-body vibration prediction is beneficial for locating potential track geometry defects with lower costs than existing methods, such as using track inspection vehicles.

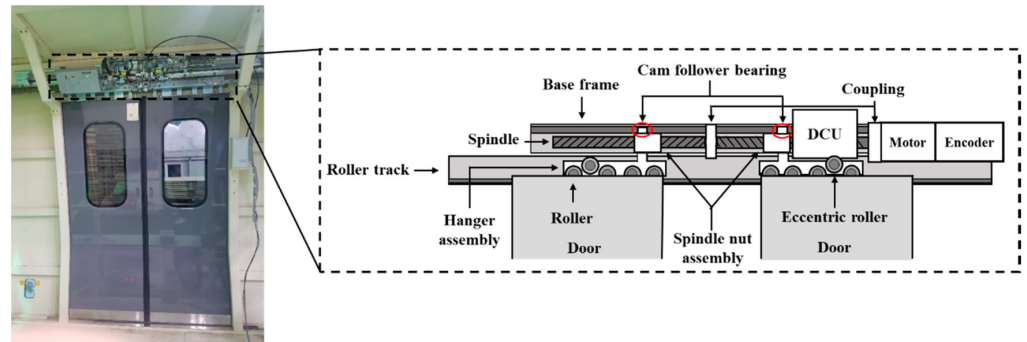
Hao et al. [66] further proposed a deep learning-based model applying attention structure and gated current unit (GRU) structure. CNN and GRU learn shape features and sequential features, respectively, and the attention structure receives the vertical, horizontal vibration, and train speed of the train as inputs, outputting the degree of vertical rail irregularity.

## 5. Train Safety

### 5.1. Train Door

Train door failures damage the train system and account for 40% of all train failure cases, leading to huge operation and maintenance expenditures. Ham et al. [67] studied a data-based approach to address train door failures. Eight failures were considered in four different scenarios. For each scenario, the change in the amount of current in the electric motor operating the train entrance was measured. Then, two techniques were used to analyze the current change data. First, 13 features were extracted from the time-series signal data using traditional feature engineering techniques based on pass filters (high and low). A KNN (k-nearest neighborhood) model detected door failures based on the extracted features. Another method is a deep learning model based on 1D

convolution. Figure 6 shows components of a train door test rigs for the experiments. The evaluation results showed that both methods showed an accuracy of 98% or more, and CNN models showed slightly higher performance, even though they used raw current signals without preprocessing.



**Figure 6.** Components of a train door test rigs. Reproduced with permission from MDPI (Creative Commons Attribution License) [67].

### 5.2. Wheel

Wheel defects in trains are also one of the main causes of damage to railway systems and railway-related facilities. Neglecting train wheel defects will shorten the service life of a railway infrastructure, which may result in unnecessary maintenance costs. Furthermore, ground vibration and noise are generated when train wheel defects are present, causing significant damage to the surrounding environment. To determine train wheel faults, Krummenacher et al. [68] focused on the vertical force of the train. They continuously measured a load of trains running at top speed from wheel load checkpoints (WLCs) placed on rails at regular intervals and studied two methods to detect train defects. The first method determined train wheel defects using an SVM model based on the train load data processed by the discrete wavelet transform (DWT), a time series data processing method. Second, a CNN-based model was developed to detect train wheel defects. They found that these proposed methods show better performance than conventional defect detection methods. In particular, the CNN-based model had strengths identifying flat spots (wheel defects that stop wheel rotation and drag along the rails) and non-roundness (wheel defects that cause vibration and noise generation).

In addition, acceleration sensors for inspecting the position of railway wheels have been widely studied to increase information utilization and efficiently perform maintenance decisions. However, the acceleration sensor has a limitation of relatively accurate detection of the longitudinal movement of train wheels but poor lateral movement accuracy. Shi et al. [69] attempted to solve this problem by utilizing an image-based point tracking method with acceleration sensor data. Their proposed model was designed based on YOLO and generated a wheel reference point indicating a wheel position from the input image and comparing it with a normal position. Furthermore, they adopted various filters and data acquisition methods to improve performance, even in weather environments such as snow and fog.

### 5.3. Suspension

Wu et al. [70] detected defects in bogie suspension components (coil spring, air spring, vertical damper, and yaw damper) by considering the increased vibration and stability of a high-speed train during accelerated operation. They developed a Bayesian deep learning-based predictive model based on accelerometer (vertical and horizontal) data collected from a bogie and accelerometer sensors attached to trains, and data with each degree of deviation of each component (vertical and horizontal). Their developed predictive model imposed perturbation by the Monte Carlo algorithm to more clearly distinguish the difference between frequent and sudden faults. Class of faults was diagnosed using drop-

out-based Bayesian deep learning. The proposed methods accurately detected rare but fatal defects, even with a small number of samples. Xie et al. [71] analyzed train vibration signals using a fast Fourier transform (FFT) that decomposed input signals by a frequency band and automated feature extraction by a deep belief network (DBN). With four different conditions (normal train, without anti-yaw shock absorber, air spring failure, and without transverse shock absorber), a total of 28,600 vibration data were collected using vibration sensors installed at various locations on a train. DBN models consisting of four restricted Boltzmann machines (RBMs) showed significant improvement in diagnosis performance.

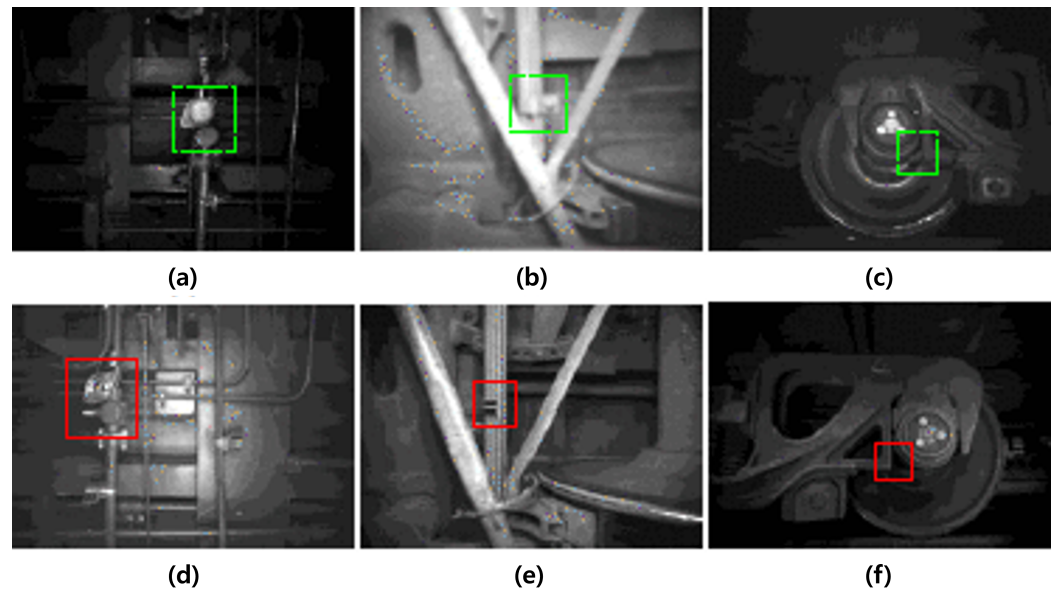
#### 5.4. Bearing

Bearing is a principal component widely used in most modern mechanical equipment. Defect inspection of bearings takes a long time and the cost of repairs is generally high, which can significantly decrease train productivity. While there have been many attempts to detect bearing defects, conventional methods have two limitations. First, methods based on features depending on expert rules or prior knowledge take too much time and human effort because different processes conducted by experts should be performed according to each specific problem. Second, traditional machine learning methods with shallow structures have had difficulty estimating nonlinear functional relationships in complex systems. In order to overcome these limitations, there have been studies to adopt deep learning to detect bearing defects.

Xu et al. [72] proposed a CNN-based model for bearing defect detection. Their proposed model used bearing vibration signals for defect detection. It converts original signals into two-dimensional features by CWT. Then, a CNN based on LeNet-5 processes features and determines its state. In addition, an ensemble method was adopted to utilize three Random Forest (RF) models with features of three specific layers as input values. He et al. [73] have developed a deep learning model that diagnoses defects using the Large Memory Storage and Retrieval Neural Network (LAMSTAR). This multi-layer fast deep learning structure can use many filters simultaneously. In addition, the short-time Fourier transform (STFT) is used to process acoustic data generated from bearings to determine when signals for each frequency band separated from the composite signal are generated. Performance tests performed in laboratory environments showed better performance than other conventional CNN models.

The features of bearing vibration signals, such as high nonlinearity, non-stationarity, and background noise, make it hard to diagnose bearing faults effectively and accurately. Zou et al. [74] proposed a deep learning method based on discrete wavelet transform (DWT) and improved DBN. First, the vibration signals from faulty bearings were converted to a two-dimensional (2D) time–frequency map. Then, the time–frequency map was processed by an improved DBN model, aiming to identify the correlation between fault features and fault types. In this way, the fault state of the bearing in the traction motor was diagnosed and identified in a semi-supervised manner.

Figure 7 shows examples of railway equipment detection. Zhan et al. [75] proposed a model that utilizes Faster R-CNN to detect the location of the target component and whether it is defective from a complex background in a bogie image. In particular, they improved the original faster R-CNN by using two layers of different sizes for extracting defect regions and enabling region of interest (ROI) pooling. Experiments on 6499 test data on four parts (cut-out cock handle, dust collector, fastening bolts, and bogie block key) showed high detection accuracy with fast speed. Sun et al. [76] proposed a CNN model that detects defects in the side frame key (SFK) and shaft bolt (SB) among bogie components. The detection model accurately located the SFK and SB from the Trouble of Running Freight Train Detection System (TFDS) image data and then cropped it to diagnose each defect.



**Figure 7.** Some typical samples of the vehicle brake system of freight trains. (a–c) Normal images. (d) Dirt collector damaged and cut-out cock handle closed (the handle of cut-out cock is not visible in a normal image). (e) The absence of fastening bolt. (f) Bogie block key missing. Reproduced with permission from IEEE [75].

Xiao et al. [77] proposed a hierarchical feature-based instance detection (HID) model to detect lost or broken defects in bogie components. Their proposed model consisted of three modules. The first module extracts hierarchical image features from train images through a CNN model. The second module delivers the extracted feature map to the region proposal network to generate a defect object area. The last module finally detects defects based on the generated regions and the feature maps. The proposed instance-level detection was evaluated on six train defects (lost pin, lost bolt, lost rivet, foreign object, broken chain, and broken wire).

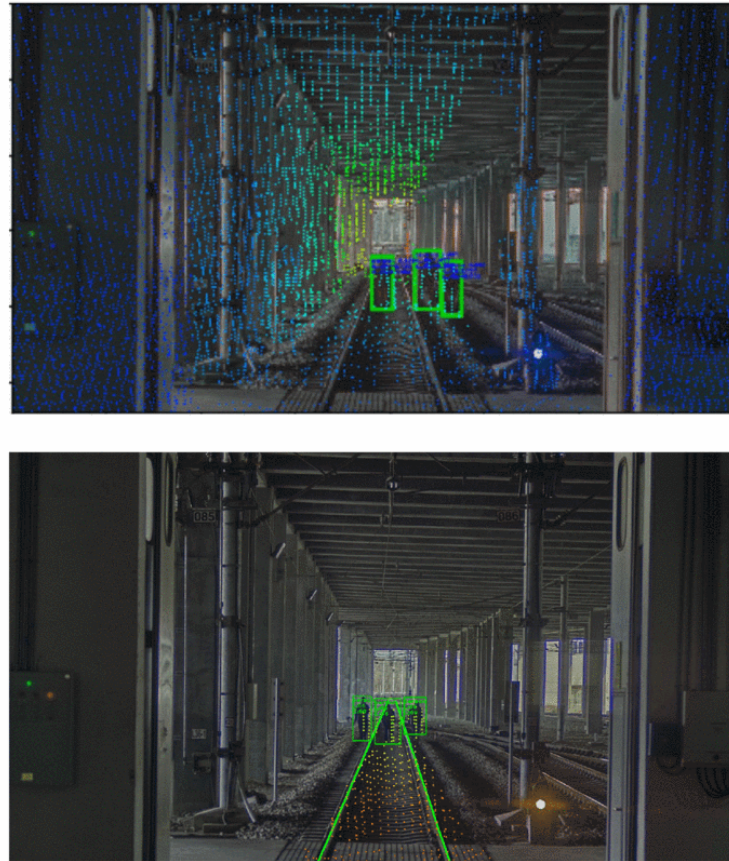
Ye et al. [78] proposed a multi-feature fusion network (MFF-net) to address the loss of small-sized areas when reducing feature map size, which results in poor detection performance. To this end, three modules were devised. First, the feature fusion method (FFM) module incorporates deep and shallow features, such as spatial location and semantic information. Second, the multi-branch dilated convolution module (MDCM), which the Inception model inspires, simultaneously enhances feature extraction around objects of different sizes. The MDCM utilizes convolution networks and multi-branch networks to accommodate multi-scale features. Finally, the squeeze and excitation block (SE) module compresses and readjusts the features to improve model representation. The proposed model outperformed other conventional models in testing with the PASCAL VOC dataset. In addition, it showed excellent stability, even for complex environmental noises.

## 6. Operation Safety

### 6.1. Railroad Trespassing

Figure 8 shows an example of railroad trespassing detection. Zaman et al. [79] proposed a deep learning framework based on mask R-CNN that automatically detects railroad trespassing in real time. The model was trained based on the COCO dataset and detects trespassing events and classifies trespasser types (car, motorcycle, truck, pedestrian, etc.). In addition, Gao et al. [80] developed a railroad trespassing detection method based on one light detection and ranging (LiDAR) system and two different focal length cameras. The cameras can provide high-resolution images and rich semantic information, while their performance can be easily affected by lighting or weather conditions, and distance estimation accuracy is limited. LiDAR can measure the distance to an object accurately and

provides a 3D image to work. However, sparse point cloud data provide limited detection capabilities in the case of small and dynamic obstacles. This work modifies an SSD network to incorporate multi-sensor data.



**Figure 8.** Example of railroad trespassing detection. Reproduced with permission from IEEE [80].

### 6.2. Railway Detection

Quickly detecting the front rail area can help prevent train accidents, such as derailments. However, railway detection in outdoor environments suffers from light-related issues, such as shadows, reflections, tunnels, and low contrast to the ground. In addition, railway detection becomes challenging in areas of overlapping multiple rails. Wang et al. [81] proposed a CNN-based deep learning model trained by the BH-rail dataset that contains railway images captured at various times. Wang et al. [82] proposed RailNet, a railway detection deep learning-based algorithm that processes video from front-view on-board cameras. RailNet consists of two networks: a network for feature extraction and another for railway segmentation. The feature extraction network has a pyramid structure to allow features to have top-to-bottom propagation. The railway segmentation network combines a ResNet50 backbone network with a fully convolutional network to generate the segmentation map.

### 6.3. Wind Risk

High-speed railways are susceptible to strong winds, which can pose a major threat to train safety. In order to ensure train safety, it is necessary to measure the wind speed of the preceding area in real time or to inform the train of the information in advance by short-term prediction. However, measured and predicted wind speed alone are not sufficient to explain wind conditions. For example, if the expected wind speed is slightly lower than the strong wind threshold, it is difficult to estimate whether a substantial wind accident can occur. Liu et al. [83] proposed a multiple attention layer based multi-instance



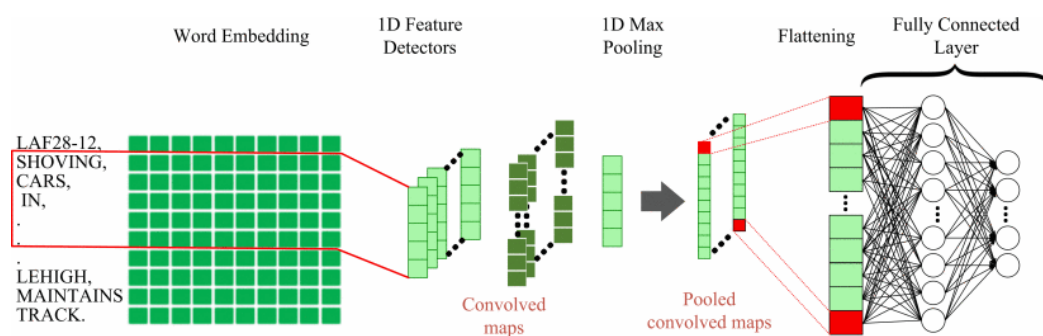
learning (MAL-MIL) model to predict substantial wind risk alongside a high-speed railway (HSR). Based on attention mechanisms and LSTM networks, the model extracted features of the future wind status and identified the relationships between the current features and strong wind incidents.

#### 6.4. Train Running Safety

There are many studies on monitoring the current state of train operation and quantifying train running safety [84–87]. However, these studies have mainly considered limited situations that can be monitored relatively simply, such as train bridges and tunnel passes. Lee et al. [88] presented a model that combines deep neural networks and recurrent neural networks for efficient train-running safety prediction. Their proposed model processed train vibration data, which was measured by an accelerometer, and predicted the wheel derail coefficient, wheel rate of lad reduction, and wheel lateral pressure. Numerical analyses were conducted using the transit simulation and the actual train-railway model, and these analysis results revealed that the proposed method has better prediction performance.

#### 6.5. Managing Accident Reports

Accident reports can help minimize risk factors to prevent future accidents. Accident reports mostly contain diverse input field entries, such as fixed field entries, which include the primary cause of accidents, or a narrative field, which is a short text description of the accident. The narratives can provide more information than a fixed field entry, but the terminologies used in the reports are not easy to understand by a non-expert reader. Heidarysafa et al. [89] applied word embedding methods, such as Word2Vec and GloVe, to narrative texts in train accident reports. As shown in Figure 9, the proposed method classifies accident cause values for the primary cause field based on embedding vectors about the narrative text. This NLP approach can help label accidents more accurately and consistently.



**Figure 9.** Proposed network structure for managing accident records. Reproduced with permission from IEEE [89].

## 7. Station Safety

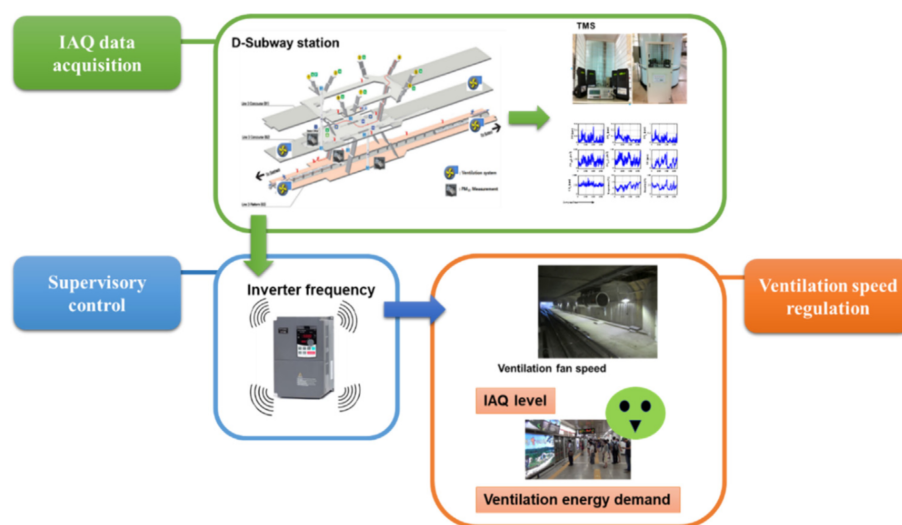
### 7.1. Accident Prevention

A railway station is dynamic and complex due to the presence of many people, including passengers and station staff, and trains that stop and depart quickly. Therefore, it is necessary to prevent and deal with various safety accidents in stations. Alawad et al. [10] proposed a model that quickly identifies three safety incidents (fall, slip, and trip). It used diverse images of platforms, escalators, and tunnels captured by CCTV in the station. The CNN-based deep learning model classified input images into two classes (fall and not fall), and it achieved a high accuracy of 82.20% and an AUC value of 82.33%.

### 7.2. Air Quality Control

Air quality measurement sensors are installed in railway stations for air quality control. However, the measurement sensors often fail due to being in the wrong location for measurement, expired sensor equipment, malfunctioning electrical equipment, etc. Since

air quality data are collected from several sensors, it is difficult to identify normal data by models having a linear or fixed structure because the variance of the data is significant, and values that do not follow a normal distribution are included. Loy-Benitez et al. [90] proposed a machine-learning-based soft sensor verification technique for detecting, diagnosing, identifying, and reconstructing abnormal measurements of multivariate air quality data. Figure 10 presents a diagram of the air quality monitoring and supervisory control process. Normal and abnormal values were extracted from the collected air quality data. A memory-gated current network auto encoder (MG-RNN-AE) algorithm based on an auto-encoder was developed to process air quality data. Furthermore, experimental results showed that the proposed method has a sustainable balance between power consumption and air quality levels, effectively performing air quality management within the station.



**Figure 10.** Diagram of the air quality monitoring and supervisory control process. Reproduced with permission from IEEE [90].

### 7.3. Simulation and Scheduling

Transportation modeling is difficult because it is a dynamic and complex system with interdependent factors, such as humans, equipment, and their temporal attributes. Recently, a deep learning approach that can extract complex high-level representations through hierarchical learning processes was applied to transportation modeling. Huang et al. [91] proposed CLF-Net, a deep learning model that combines 3D-CNN, LSTM, and fully connected neural networks to handle complex variables in dynamic systems. The proposed model separately processes data with different attributes for better predictive performance, uses spatio-temporal variables to capture space-time dependencies, and receives variables to learn the potential effects of static factors.

With the development of cities, short-term traffic prediction has become the core of the intelligence transportation system (ITS). Accurate short-term traffic forecasting can provide technical support to monitor train passenger flow and warn of excessive traffic congestion. Tang et al. [92] proposed a spatio-temporal long-term network (ST-LSTM) that captures spatio-temporal features from railway traffic data. Their proposed model improved the original LSTM structure, focusing on temporal rather than spatial features.

Predicting train delays can improve the quality of train operation, which helps to estimate train operation and more accurately make reasonable operational decisions. A train delay is affected by many factors, such as passenger flow, failure, extreme weather, and dispatch strategies. Considering such temporal and spatial factors between multiple trains and routes is challenging, which makes it difficult to accurately predict train delays. Zhang et al. [93] focused on predicting the cumulative effects of train delays over a certain period of time, represented by the total number of arrival delays in one station, rather than predicting each specific delay time of a single train. A deep learning framework based on

the spatio-temporal attention mechanism and spatio-temporal convolution was proposed. Their model receives recent input of daily and weekly time series data and each component includes a spatio-temporal attention mechanism and spatio-temporal convolution, which can effectively capture spatio-temporal characteristics. Experiments on train operation data in the railway passenger ticket system of China demonstrated that the proposed model clearly outperforms existing performance criteria in train delay prediction.

## 8. Discussion and Conclusions

Our literature survey shows that artificial intelligence has been widely applied to various railway safety issues, such as railway infrastructure, trains, operations, and stations. This review details both opportunities and challenges for artificial intelligence in railway safety.

First of all, advances in data-driven artificial technologies can improve conventional railway safety performance methods. In addition, many studies have shown the feasibility of automating or supplementing conventional railway safety inspection procedures that depend on visual analysis or domain knowledge of a human expert. The proposed model structures in the discussed studies were determined based on the input data types. An image or video is one of the most common data types in artificial intelligence applications for railway safety. Many studies for defect detection (e.g., catenary and rail surface defects) developed CNN-based deep learning models and train vibration is another popular data source for railway safety. Accelerometers can easily measure train vibrations and LSTM-based models have been used to extract unique patterns from accelerometer data.

On the other hand, there are also challenging issues in utilizing artificial intelligence for railway safety that further studies should consider. We divided the addressed issues into two categories: (1) performance optimization and (2) generalization. First, many studies addressed the necessity of further performance improvement in artificial intelligence. For example, model accuracy needs to be improved to reach practical requirements or the model structure should be more optimized to be executed in real-time. Second, generalization of the proposed methods was issued by many studies. Some studies used simulation data in a lab setting, so in-situ validation needs to be performed for practical application.

More details regarding research issues in artificial intelligence for railway safety, addressed by prior studies, are explained in subsequent subsections.

### 8.1. Performance Optimization

#### 8.1.1. Dealing with a Lack of Data

Developing deep learning models for railway safety is challenged by practical limitations of data volume or quality, such as diverse noises in railway environments and insufficient labeled data. Therefore, it is necessary to deal with such data deficiencies when developing artificial intelligence for railway safety. For example, Xiao et al. [77] utilized a hierarchy of features for training a deep learning model with a small number of labeled data. Ensemble methods that integrate different machine learning algorithms can help increase the efficiency of model learning. For example, Xu et al. [72] considered an ensemble method that integrates a CNN-based model and RF for bearing fault diagnosis. The ensemble approach can be efficient with a relatively small number of data rather than the end-to-end deep learning approach. In addition, unsupervised models can help deal with a small number of labeled data. Soares et al. [63] expected to improve system performance by analyzing other clustering algorithms or adjusting internal parameters.

#### 8.1.2. Processing Time

Beyond model accuracy, processing time can be one of the essential requirements in artificial intelligence for railway safety. In particular, real-time processing can be required for high-speed train applications. For example, Wang et al. [81] suggested further studies to develop a real-time system that recognizes moving obstacles by combining railway area

recognition and obstacle detection steps. In addition, Lee et al. [88] expected that their system could be utilized for real-time train control to reduce the risk of train derailment.

### 8.1.3. New Data Source

Most prior studies were conducted with a limited data source. For example, Wang et al. [40] emphasized the necessity of improvement of data quality. Their data were collected in limited circumstances, such as fixing the camera angle when taking the image data. In addition, Jiang et al. [8] argued that experiments in various railway conditions, such as the angle or length of the rail, should be conducted to develop a fault detection system. Furthermore, Ma et al. [65], who developed a method for rail defect inspection based on vibration signals, commented that considering various train types and driving speeds can help improve performance.

While most studies about artificial intelligence for railway safety have utilized image or vibration data, some studies have explored other data sources to improve model performance. For example, Wolf et al. [94] proposed using LiDAR sensor data to understand situations and components in 3D railway images. Suwansin and Phasukkit [9] utilized acoustic emission signals from rails for rolling contact fatigue. Furthermore, artificial intelligence could be improved and optimized by harnessing various situational features in the railway domain. Krummenacher et al. [68] developed an efficient model for detecting machine-learning-based train wheel defects by additionally considering the exterior characteristics of the train.

## 8.2. Generalization

### 8.2.1. Tasks

Many prior studies have proposed a deep learning framework for defect detection in railways. However, the proposed frameworks were developed and evaluated with certain types of defects and there is much room for improvement to satisfy practical requirements. For example, Chandran et al. [62] focused on one fastener type and addressed the need to study the feasibility of the proposed method for other types. Similarly, Akhila et al. [4] also noted that the proposed framework needs to be improved with other examples and under different contexts. Wu et al. [70] conducted a study to detect defects in truck joints and accessories, further noting that a partial defect detection study should also be conducted to eliminate potential risk factors for train operation.

### 8.2.2. Validation with In-Situ Data

Because in-situ data acquisition is challenging in railways, many studies have been conducted with artificial data acquired in lab experiments. Even though the models trained by lab-setting datasets can ensure feasibility of the proposed methods and provide initial insights, these studies have addressed the need for further research with actual train-running data for validation. Shebani and Iwinicki [52] performed laboratory testing under limited conditions and noted that validation of the developed method in the field is necessary. Similarly, Kim et al. [5] addressed a gap between an actual train situation and simulation data. Shi et al. [69] developed a model to monitor rail-track geometry defects but reported that the model performance decreased in harsher outdoor situations. Unexpected noises can also cause such decreases in field performance [60]. Additionally, actual data can contain more diverse and complex conditions that are rarely covered by lab experiments. Ham et al. [67] detected a train entrance door failure using data generated by manipulating doors with several abnormal conditions, so their model should be further studied using actual train door failure data.

**Author Contributions:** Literature review, data analysis, and writing—original draft preparation, K.O., N.J., J.K., J.S. and H.J.; writing—review and editing, M.Y. and M.K.; visualization, N.J. and J.K.; supervision, M.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was partly supported by a grant from R&D Program (Development of monitoring system using InSAR satellite information data, PK2203B3) of the Korea Railroad Research Institute and partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2022-00155885, Artificial Intelligence Convergence Innovation Human Resources Development (Hanyang University ERICA)).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

Table A1. Studies of AI applications for railway safety.

Ref	Safety Type	Category	Target	Data Type	Source	Training Data	Test Data	Method	Performance
[37]	Railway Infra	Catenary	Insulator	Image	Custom	12,000	6000	Faster R-CNN, CNN, AE (Auto-Encoder)	0.95 (F1-score)
[38]	Railway Infra	Catenary	Dropper	Image	PASCAL VOC <sup>1</sup> , MSCOCO <sup>2</sup>	1172	293	Faster R-CNN, FPN (Feature Pyramid Network), ResNet	0.87 (mAP@0.5), 0.84 (mAP@0.7)
[39]	Railway Infra	Catenary	Clevis	Image	PASCAL VOC <sup>1</sup>	4000 (5075 clevis)	2000 (2563 clevis)	Faster R-CNN, CNN	0.76–0.97 (Accuracy)
[40]	Railway Infra	Catenary	Split pin	Image	Custom	8256 (66,259 split pins)	2670 (21,472 split pins)	YOLOv3, DeepLab v3+	0.99 (Accuracy)
[44]	Railway Infra	Catenary	Current-carrying ring	Image	Custom	3050	1500	Attention, RetinaNet	0.70 (mAP@0.5)
[45]	Railway Infra	Surface	Scouring, Breakage, Corrugation, Headcheck	Video	Custom	Unknown	Unknown	PCA, SVD, RF (Random Forest)	0.85–0.98 (Accuracy)
[47]	Railway Infra	Surface	Defect	Image	PASCAL VOC <sup>1</sup>	184	11	YOLOv3, ResNet	0.97–1 (Detection Rate), 0.15 s (Time Cost)
[48]	Railway Infra	Surface	Defect	Image	Custom	142,416	9494	MobileNetV2, YOLOv3	0.87 (mAP)
[5]	Railway Infra	Surface	Defect	Image	Custom	1905	211	CNN, VGG19 <sup>3</sup>	0.92–0.92 (F1-score)
[50]	Railway Infra	Surface	Defect	Image	Custom	120	7	SegNet	1 (Detection Rate), 0.99 (Accuracy)
[46]	Railway Infra	Surface	Defect	Image	Custom	2916	324	CNN	0.90–0.91 (F1-score), 1.00–2.03 s (time cost)
[49]	Railway Infra	Surface	Defect	Image	Custom	5793	1517	Inception3, CNN	0.92 (Recall), 0.92 (Precision)
[8]	Railway Infra	Surface	Rolling Contact Fatigue	Signal (Laser ultrasonic)	Custom	Unknown	256	SVM	0.99 (Accuracy)
[51]	Railway Infra	Surface	Squat	Signal (Acceleration)	Custom	819	204	CVAE (Convolutional Variational Auto Encoder)	0.93–0.97 (Accuracy)
[9]	Railway Infra	Surface	Crack	Signal (Acoustic emission)	Custom	360	90	DNN	0.77 (Accuracy)
[52]	Railway Infra	Surface	Wear	Measurements (Load, Yaw angle, Speed, Wheel, Rail profile)	Custom	182	39	ANN	0.81–0.93 (Accuracy)
[53]	Railway Infra	Surface	Defect	Image	Custom	540	60	CNN, ResNet	0.93–0.97 (F-measure)
[54]	Railway Infra	Surface	Defect	Image	Custom	146	49	1D-CNN, LSTM	0.93–0.94 (Recall), 0.84–0.92 (Precision), 0.88–0.93 (F1-Score)
[56]	Railway Infra	Surface	Dust, Oil	Signal (3D Laser camera)	Custom	7500	2500	CNN	0.98 (Accuracy)
[57]	Railway Infra	Surface	Degradation	Signal (Acceleration)	Melbourne Tram Network Data	Unknown	Unknown	SVM, RF, ANN	0.71–0.78 (Adjusted R <sup>2</sup> ), 0.80–0.91 (RMSE)
[58]	Railway Infra	Components	Spike, Clip, Tie Plate	Image	Custom <sup>4</sup>	800	200	YOLACT, Res2Net, ResNet	0.60–0.64 (mAP)
[59]	Railway Infra	Components	Fastener, Crosstie, Ballast, Gage	Image	Custom	650,518	162,629	CNN	0.95 (Accuracy)

Table A1. Cont.

Ref	Safety Type	Category	Target	Data Type	Source	Training Data	Test Data	Method	Performance
[60]	Railway Infra	Components	Settlement, Dipped joint	Signal (Acceleration)	Custom	1155	495	DNN, CNN, RNN	0.84–0.99 (Accuracy)
[7]	Railway Infra	Components	Joint, Crossing, Turnout	Signal (Acceleration)	Custom	23	41	CNN, ResNet	0.99 (Accuracy)
[61]	Railway Infra	Components	Joint	Signal (Acceleration)	Custom	129	295	CNN, ResNet	0.74–0.91 (F1-score)
[62]	Railway Infra	Components	Clamp	Signal (Eddy current)	Custom	2076	890	SVM, k-NN, RF	0.97 (Precision), 0.96 (Recall)
[63]	Railway Infra	Components	Rail Switch Machine	Signal (Electric current)	Custom	Unknown	615	K-means clustering	0.86 (Silhouette score)
[64]	Railway Infra	Components	Rail Slab	Signal (Vibration)	Custom	1774	760	RF	0.96 (Accuracy)
[65]	Railway Infra	Geometry	Quality	Signal (Vibration)	(Comprehensive Inspection Train) Custom	5,400,000	600,000	CNN, LSTM	0.005–0.006 (MAE), 0.007–0.008 (RMSE)
[66]	Railway Infra	Geometry	Irregularity	Signal (Acceleration)	(Beijing–Shanghai, Beijing–Guangzhou and Nanjing–Hangzhou HSRs) Custom	200 km	100 km	Attention, CNN, GRU	0.25–0.51 (MAE), 0.33–0.66 (RMSE)
[67]	Train	Door	Defect	Signal (Current)	Custom	440	186	CNN, k-NN	0.98–0.99 (Accuracy)
[68]	Train	Wheel	Defect	Signal (Vertical force)	Custom	7860	2565	DNN, SVM	0.81–0.89 (Accuracy)
[69]	Train	Wheel	Displacement	Image	Custom <sup>5</sup>	2301	767	CNN, YOLOv3 <sup>6</sup>	0.35 (Miss Detection Rate)
[70]	Train	Suspension	Coil Spring, Air Spring, Vertical Damper, Lateral Damper, Yaw Damper	Signal (Vibration)	Case Western Reserve University (CWRU) Bearing Data Center	59,520	7440	Bayesian DL	0.77–0.99 (AUROC)
[71]	Train	Suspension	Anti-yaw Shock Absorber, Air Spring, Transverse Shock Absorber	Signal (Vibration)	Custom	14000	208	DBN (Deep Belief Network)	0.23–0.54 (Accuracy)
[72]	Train	Bearing	Defect	Signal (Vibration)	Case Western Reserve University (CWRU) Bearing Data Center <sup>7</sup>	2000	2000	CNN, RF (Random Forest), LeNet-5	0.97 (Accuracy)
[73]	Train	Bearing	Defect	Signal (Acoustic emission)	Custom	270	180	DNN	0.96–1 (Accuracy)
[74]	Train	Bearing	Defect	Signal (Vibration)	Custom	640	160	DBN (Deep Belief Network)	0.95 (Accuracy)
[75]	Train	Other Components	Cut-out cock handle, Dust collector, Fastening bolt, Bogie block key	Image	PASCAL VOC <sup>1</sup>	8794	6493	Faster R-CNN, CNN	0.98–1 (Correct Detection Rate)
[76]	Train	Other Components	Side Frame Key, Shaft Bolt	Image	PASCAL VOC <sup>1</sup>	2321	354	CNN	0.93–1 (Accuracy)
[77]	Train	Other Components	Bolt, Pin, Rivet, Chain, Wire	Image	Custom	307	72	CNN, ResNet	0.90 (Recall), 0.86 (Precision), 0.88 (F1-score)
[78]	Train	Other Components	Bolt, Retaining key	Image	Custom	3614	903	SSD, CNN	0.89(mAP)
[79]	Operation	Railroad trespassing	Trespasser	Video	Custom	Unknown	69 h	Mask R-CNN	Unknown

Table A1. Cont.

Ref	Safety Type	Category	Target	Data Type	Source	Training Data	Test Data	Method	Performance
[80]	Operation	Railroad Trespassing	Obstacle	Image (Camera, LiDAR)	Custom	Unknown	Unknown	SSD	0.05–0.21 (Error Rate)
[81]	Operation	Railway Detection	Railway area	Image	Custom (Beijing metro Yanfang line and Shanghai metro line 6)	4494	1123	CNN	0.99 (MIoU), 0.99 (Mean Pixel Accuracy)
[82]	Operation	Railway Detection	Railway area	Image	Custom	2500	300	ResNet50	0.92 (Accuracy), 0.90 (mIoU), 0.87 (F1-score)
[83]	Operation	Wind Risk	Wind Speed	Wind Speed	Custom (Beijing-Shanghai HSR)	23,792	9517	Attention, LSTM	0.82 (AUC), 0.95 (F1-score)
[88]	Operation	Train Running Safety	Wheel Derail Coefficient, Wheel Rate of Lad Reduction, Wheel Lateral Pressure	Signal (Vibration)	Custom	9600	2400	DNN, LSTM	0.42 (RMSE)
[89]	Operation	Managing Accident Reports	Accident Narrative	Accident Narrative Documents	Federal Railroad Administration (FRA) reports <sup>8</sup>	None (Pre-trained Model)	40,164	CNN, LSTM, GRU <sup>9</sup>	0.57–0.65 (F1-score)
[10]	Station	Accident Prevention	Fall, Slip, Trip	Video	Custom & Le2i Dataset	10,459	1307	CNN	0.72–0.82 (Accuracy)
[90]	Station	Air Quality Control	Air Quality	NO, NO <sub>2</sub> , NO <sub>x</sub> , PM <sub>10</sub> , PM <sub>2.5</sub> , CO, and CO <sub>2</sub> , Temperature, and Humidity	Custom	504	168	MG-RNN (Memory-Gated RNN), AE	1.74–15.01 (RMSE)
[91]	Station	Simulation and Scheduling	Dynamic System	Train Operation Record	Custom	171,990	57,330	3D-CNN, LSTM <sup>10</sup>	0.63–0.87 (RMSE), 0.44–0.51 (MAE)
[92]	Station	Simulation and Scheduling	Transportation Flow	Card Records	Custom (Chongqing City Transportation Development & Investment Group)	4,800,000	1,200,000	LSTM	5.72 (RMSE), 4.41 (MAE)
[93]	Station	Simulation and Scheduling	Delay	Train Operation Record	China Railway Passenger Ticket System	Unknown	Unknown	Attention, CNN	0.16 (MAE), 0.45 (RMSE)

<sup>1</sup> <http://host.robots.ox.ac.uk/pascal/VOC/> (accessed on 29 September 2022), <sup>2</sup> <https://cocodataset.org/#download> (accessed on 29 September 2022), <sup>3</sup> [https://github.com/lottopotato/railroad\\_surface\\_defect\\_segmentation.git](https://github.com/lottopotato/railroad_surface_defect_segmentation.git) (accessed on 29 September 2022), <sup>4</sup> [https://github.com/jonguo111/Rail\\_components\\_image\\_data.git](https://github.com/jonguo111/Rail_components_image_data.git) (accessed on 29 September 2022), <sup>5</sup> <http://doi.org/10.14279/depositonce-12223> (accessed on 29 September 2022), <sup>6</sup> <https://github.com/quickhdsdc/Point-Tracking-for-Displacement-Measurement-in-Railway-Applications.git> (accessed on 29 September 2022), <sup>7</sup> <https://engineering.case.edu/bearingdatacenter/download-data-file> (accessed on 29 September 2022), <sup>8</sup> <https://railroads.dot.gov/safety-data/forms-guides-publications/dataset-downloads> (accessed on 29 September 2022), <sup>9</sup> [https://github.com/Heidarysafa/train\\_accidents.git](https://github.com/Heidarysafa/train_accidents.git) (accessed on 29 September 2022), <sup>10</sup> [https://github.com/PingH129/A-deep-learning-approach-for-multi-attribute-data\\_-A-study-of-train-delay-prediction-in-railway-syst.git](https://github.com/PingH129/A-deep-learning-approach-for-multi-attribute-data_-A-study-of-train-delay-prediction-in-railway-syst.git) (accessed on 29 September 2022).



**Table A2.** Performance metrics.

Category	Name	Formula	Description
Classification	Accuracy	$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$	Fraction of the total samples that were correctly classified
	Recall	$Recall = \frac{T_p}{T_p + F_n}$	Fraction of the number of true positives ( $T_p$ ) over the number of true positives plus the number of false negatives ( $F_n$ )
	Precision	$Precision = \frac{T_p}{T_p + F_p}$	Fraction of the number of true positives ( $T_p$ ) over the number of true positives plus the number of false positives ( $F_p$ )
	mAP (mean Average Precision)	$mAP = \frac{1}{n} \sum_{i=1}^n AP_i$	Average Precision (AP): Area under the precision-recall curve above mean Average Precision(mAP): Mean of all the AP
	F1-score	$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$	Harmonic mean of <i>precision</i> and <i>recall</i>
	AUROC (Area under ROC)	$AUROC = \int_0^1 ROC(x) dx$	The entire two-dimensional area underneath the ROC curve from (0,0) to (1,1)
	Silhouette score	$S = \frac{(nearest - intra)}{\max(intra, nearest)}$	<i>intra</i> : Mean distance between the observation and all other data points in the same cluster <i>nearest</i> : Mean distance between the observation and all other data points of the next nearest cluster
Regression	Adjusted $R^2$	$Adjusted R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$	Percentage of variance in the target field that is explained by the input. $R^2$ = Sample R-squared $N$ = Total Sample Size $p$ = Number of independent variable
	RMSE (Root Mean Squared Error)	$RMSE = \sqrt{MSE}$	$MSE$ : Difference between the predicted and observed values in model $RMSE$ : Square root of the $MSE$
	MAE (Mean Absolute Error)	$MAE = \frac{1}{n} \sum_{i=1}^n  x_i - x $	Mean of absolute difference between model prediction and target value
Segmentation	mIoU (mean Intersection over Union)	$mIoU = \frac{\sum_{i=1}^n IoU_i}{n}$	Average between the <i>IoU</i> (Intersection over Union) of the segmented objects over all the images of the test dataset $IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}$
	mPA (mean Pixel Accuracy)	$mPA = \frac{1}{k} \sum_{j=1}^k \frac{n_{jj}}{t_j}$	$n_{jj}$ : Total number of pixels both classified and labeled as class $j$ $t_j$ : Total number of pixels labeled as class $j$

## References

1. Nilsson, N.J.; Nilsson, N.J. *Artificial Intelligence: A New Synthesis*; Morgan Kaufmann: San Francisco, CA, USA, 1998.
2. McCarthy, J. What Is Artificial Intelligence. 2004. Available online: <http://www-formal.stanford.edu/jmc/whatisai.html> (accessed on 29 September 2022).
3. Boden, M. Artificial intelligence and natural man. *Synthese* **1980**, *43*, 433–451.
4. Akhila, C.; Diamond, C.A.; Poonia, A.M. Convolutional Neural network based Online Rail surface Crack Detection. In Proceedings of the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 6–8 May 2021; pp. 1602–1606.
5. Kim, H.; Lee, S.; Han, S. Railroad Surface Defect Segmentation Using a Modified Fully Convolutional Network. *KSII Trans. Internet Inf. Syst. TIIIS* **2020**, *14*, 4763–4775.
6. Tsunashima, H. Condition monitoring of railway tracks from car-body vibration using a machine learning technique. *Appl. Sci.* **2019**, *9*, 2734. [[CrossRef](#)]
7. Yang, C.; Sun, Y.; Ladubec, C.; Liu, Y. Developing machine learning-based models for railway inspection. *Appl. Sci.* **2020**, *11*, 13. [[CrossRef](#)]
8. Jiang, Y.; Wang, H.; Tian, G.; Yi, Q.; Zhao, J.; Zhen, K. Fast classification for rail defect depths using a hybrid intelligent method. *Optik* **2019**, *180*, 455–468. [[CrossRef](#)]
9. Suwansin, W.; Phasukkit, P. Deep learning-based acoustic emission scheme for nondestructive localization of cracks in train rails under a load. *Sensors* **2021**, *21*, 272. [[CrossRef](#)]
10. Alawad, H.; Kaewunruen, S.; An, M. A deep learning approach towards railway safety risk assessment. *IEEE Access* **2020**, *8*, 102811–102832. [[CrossRef](#)]
11. Tang, R.; De Donato, L.; Besinović, N.; Flammini, F.; Goverde, R.M.; Lin, Z.; Liu, R.; Tang, T.; Vittorini, V.; Wang, Z. A literature review of Artificial Intelligence applications in railway systems. *Transp. Res. Part C Emerg. Technol.* **2022**, *140*, 103679. [[CrossRef](#)]
12. Liu, S.; Wang, Q.; Luo, Y. A review of applications of visual inspection technology based on image processing in the railway industry. *Transp. Saf. Environ.* **2019**, *1*, 185–204. [[CrossRef](#)]
13. Ghofrani, F.; He, Q.; Goverde, R.M.; Liu, X. Recent applications of big data analytics in railway transportation systems: A survey. *Transp. Res. Part C Emerg. Technol.* **2018**, *90*, 226–246. [[CrossRef](#)]
14. Hu, W.; Wang, W.; Ai, C.; Wang, J.; Wang, W.; Meng, X.; Liu, J.; Tao, H.; Qiu, S. Machine vision-based surface crack analysis for transportation infrastructure. *Autom. Constr.* **2021**, *132*, 103973. [[CrossRef](#)]
15. Sedghi, M.; Kauppila, O.; Bergquist, B.; Vanhatalo, E.; Kulahci, M. A taxonomy of railway track maintenance planning and scheduling: A review and research trends. *Reliab. Eng. Syst. Saf.* **2021**, *215*, 107827. [[CrossRef](#)]
16. Yin, M.; Li, K.; Cheng, X. A review on artificial intelligence in high-speed rail. *Transp. Saf. Environ.* **2020**, *2*, 247–259. [[CrossRef](#)]
17. Wen, C.; Huang, P.; Li, Z.; Lessan, J.; Fu, L.; Jiang, C.; Xu, X. Train dispatching management with data-driven approaches: A comprehensive review and appraisal. *IEEE Access* **2019**, *7*, 114547–114571. [[CrossRef](#)]
18. Chenariyan Nakhaee, M.; Hiemstra, D.; Stoelinga, M.; Noort, M.v. The recent applications of machine learning in rail track maintenance: A survey. In Proceedings of the International Conference on Reliability, Safety, and Security of Railway Systems, Lille, France, 4–6 June 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 91–105.
19. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
21. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA 21–26 July 2017; pp. 4700–4708.
22. Gao, S.H.; Cheng, M.M.; Zhao, K.; Zhang, X.Y.; Yang, M.H.; Torr, P. Res2net: A new multi-scale backbone architecture. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 652–662. [[CrossRef](#)]
23. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
24. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
25. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
26. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
27. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
28. Bolya, D.; Zhou, C.; Xiao, F.; Lee, Y.J. Yolact: Real-time instance segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9157–9166.

29. Mika, S.; Ratsch, G.; Weston, J.; Scholkopf, B.; Mullers, K.R. Fisher discriminant analysis with kernels. In Proceedings of the Neural Networks for Signal Processing IX: 1999 IEEE Signal Processing Society Workshop (cat. no. 98th8468), Madison, WI, USA, 25 August 1999; pp. 41–48.
30. Hardoon, D.R.; Szedmak, S.; Shawe-Taylor, J. Canonical correlation analysis: An overview with application to learning methods. *Neural Comput.* **2004**, *16*, 2639–2664. [[CrossRef](#)]
31. Golub, G.H.; Van Loan, C.F. *Matrix Computations*; JHU Press: Baltimore, MD, USA, 2013.
32. Tenenbaum, J.B.; Silva, V.d.; Langford, J.C. A global geometric framework for nonlinear dimensionality reduction. *Science* **2000**, *290*, 2319–2323. [[CrossRef](#)]
33. Roweis, S.T.; Saul, L.K. Nonlinear dimensionality reduction by locally linear embedding. *Science* **2000**, *290*, 2323–2326. [[CrossRef](#)]
34. Kramer, M.A. Nonlinear principal component analysis using autoassociative neural networks. *AIChE J.* **1991**, *37*, 233–243. [[CrossRef](#)]
35. Fischer, A.; Igel, C. Training restricted Boltzmann machines: An introduction. *Pattern Recognit.* **2014**, *47*, 25–39. [[CrossRef](#)]
36. Hinton, G.E. Deep belief networks. *Scholarpedia* **2009**, *4*, 5947. [[CrossRef](#)]
37. Kang, G.; Gao, S.; Yu, L.; Zhang, D. Deep architecture for high-speed railway insulator surface defect detection: Denoising autoencoder with multitask learning. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 2679–2690. [[CrossRef](#)]
38. Guo, Q.; Liu, L.; Xu, W.; Gong, Y.; Zhang, X.; Jing, W. An improved faster R-CNN for high-speed railway dropper detection. *IEEE Access* **2020**, *8*, 105622–105633. [[CrossRef](#)]
39. Han, Y.; Liu, Z.; Lyu, Y.; Liu, K.; Li, C.; Zhang, W. Deep learning-based visual ensemble method for high-speed railway catenary clevis fracture detection. *Neurocomputing* **2020**, *396*, 556–568. [[CrossRef](#)]
40. Wang, J.; Luo, L.; Ye, W.; Zhu, S. A defect-detection method of split pins in the catenary fastening devices of high-speed railway based on deep learning. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9517–9525. [[CrossRef](#)]
41. Roy Choudhury, A.; Vanguri, R.; Jambawalikar, S.R.; Kumar, P. Segmentation of brain tumors using DeepLabv3+. In Proceedings of the International MICCAI Brainlesion Workshop, Granada, Spain, 16 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 154–167.
42. Kurth, T.; Treichler, S.; Romero, J.; Mudigonda, M.; Luehr, N.; Phillips, E.; Mahesh, A.; Matheson, M.; Deslippe, J.; Fatica, M.; et al. Exascale deep learning for climate analytics. In Proceedings of the SC18: International Conference for High Performance Computing, Networking, Storage and Analysis, Dallas, TX, USA, 11–16 November 2018; pp. 649–660.
43. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
44. Chen, Y.; Song, B.; Zeng, Y.; Du, X.; Guizani, M. Fault diagnosis based on deep learning for current-carrying ring of catenary system in sustainable railway transportation. *Appl. Soft Comput.* **2021**, *100*, 106907. [[CrossRef](#)]
45. Santur, Y.; Karaköse, M.; Akin, E. Random forest based diagnosis approach for rail fault inspection in railways. In Proceedings of the 2016 National Conference on Electrical, Electronics and Biomedical Engineering (ELECO), Bursa, Turkey, 1–3 December 2016; pp. 745–750.
46. Faghih-Roohi, S.; Hajizadeh, S.; Núñez, A.; Babuska, R.; De Schutter, B. Deep convolutional neural networks for detection of rail surface defects. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 2584–2589.
47. Yanan, S.; Hui, Z.; Li, L.; Hang, Z. Rail surface defect detection method based on YOLOv3 deep learning networks. In Proceedings of the 2018 Chinese Automation Congress (CAC), Xi'an, China, 30 November–2 December 2018; pp. 1563–1568.
48. Yuan, H.; Chen, H.; Liu, S.; Lin, J.; Luo, X. A deep convolutional neural network for detection of rail surface defect. In Proceedings of the 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), Hanoi, Vietnam, 14–17 October 2019; pp. 1–4.
49. Shang, L.; Yang, Q.; Wang, J.; Li, S.; Lei, W. Detection of rail surface defects based on CNN image recognition and classification. In Proceedings of the 2018 20th International Conference on Advanced Communication Technology (ICACT), Chuncheon, Korea, 11–14 February 2018; pp. 45–51.
50. Liang, Z.; Zhang, H.; Liu, L.; He, Z.; Zheng, K. Defect detection of rail surface with deep convolutional neural networks. In Proceedings of the 2018 13th World Congress on Intelligent Control and Automation (WCICA), Changsha, China, 4–8 July 2018; pp. 1317–1322.
51. Yuan, Z.; Zhu, S.; Chang, C.; Yuan, X.; Zhang, Q.; Zhai, W. An unsupervised method based on convolutional variational auto-encoder and anomaly detection algorithms for light rail squat localization. *Constr. Build. Mater.* **2021**, *313*, 125563. [[CrossRef](#)]
52. Shebani, A.; Iwnicki, S. Prediction of wheel and rail wear under different contact conditions using artificial neural networks. *Wear* **2018**, *406*, 173–184. [[CrossRef](#)]
53. Wu, Y.; Qin, Y.; Qian, Y.; Guo, F.; Wang, Z.; Jia, L. Hybrid deep learning architecture for rail surface segmentation and surface defect detection. *Comput.-Aided Civ. Infrastruct. Eng.* **2022**, *37*, 227–244. [[CrossRef](#)]
54. Zhang, D.; Song, K.; Wang, Q.; He, Y.; Wen, X.; Yan, Y. Two deep learning networks for rail surface defect inspection of limited samples with line-level label. *IEEE Trans. Ind. Inform.* **2020**, *17*, 6731–6741. [[CrossRef](#)]
55. Hajizadeh, S.; Núñez, A.; Tax, D.M. Semi-supervised rail defect detection from imbalanced image data. *IFAC-PapersOnLine* **2016**, *49*, 78–83. [[CrossRef](#)]

56. Santur, Y.; Karaköse, M.; Akin, E. A new rail inspection method based on deep learning using laser cameras. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 16–17 September 2017; pp. 1–6.
57. Falamarzi, A.; Moridpour, S.; Nazem, M. Development of a tram track degradation prediction model based on the acceleration data. *Struct. Infrastruct. Eng.* **2019**, *15*, 1308–1318. [[CrossRef](#)]
58. Guo, F.; Qian, Y.; Wu, Y.; Leng, Z.; Yu, H. Automatic railroad track components inspection using real-time instance segmentation. *Comput.-Aided Civ. Infrastruct. Eng.* **2021**, *36*, 362–377. [[CrossRef](#)]
59. Gibert, X.; Patel, V.M.; Chellappa, R. Deep multitask learning for railway track inspection. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 153–164. [[CrossRef](#)]
60. Sresakoolchai, J.; Kaewunruen, S. Detection and severity evaluation of combined rail defects using deep learning. *Vibration* **2021**, *4*, 341–356. [[CrossRef](#)]
61. Sun, Y.; Liu, Y.; Yang, C. Railway joint detection using deep convolutional neural networks. In Proceedings of the 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), Vancouver, BC, Canada, 22–26 August 2019; pp. 235–240.
62. Chandran, P.; Thierry, F.; Odelius, J.; Famurewa, S.M.; Lind, H.; Rantatalo, M. Supervised Machine Learning Approach for Detecting Missing Clamps in Rail Fastening System from Differential Eddy Current Measurements. *Appl. Sci.* **2021**, *11*, 4018. [[CrossRef](#)]
63. Soares, N.; de Aguiar, E.P.; Souza, A.C.; Goliatt, L. Unsupervised machine learning techniques to prevent faults in railroad switch machines. *Int. J. Crit. Infrastruct. Prot.* **2021**, *33*, 100423. [[CrossRef](#)]
64. Guo, G.; Cui, X.; Du, B. Random-Forest Machine Learning Approach for High-Speed Railway Track Slab Deformation Identification Using Track-Side Vibration Monitoring. *Appl. Sci.* **2021**, *11*, 4756. [[CrossRef](#)]
65. Ma, S.; Gao, L.; Liu, X.; Lin, J. Deep learning for track quality evaluation of high-speed railway based on vehicle-body vibration prediction. *IEEE Access* **2019**, *7*, 185099–185107. [[CrossRef](#)]
66. Hao, X.; Yang, J.; Yang, F.; Sun, X.; Hou, Y.; Wang, J. Track geometry estimation from vehicle-body acceleration for high-speed railway using deep learning technique. *Veh. Syst. Dyn.* **2022**, 1–21. [[CrossRef](#)]
67. Ham, S.; Han, S.Y.; Kim, S.; Park, H.J.; Park, K.J.; Choi, J.H. A comparative study of fault diagnosis for train door system: traditional versus deep learning approaches. *Sensors* **2019**, *19*, 5160. [[CrossRef](#)] [[PubMed](#)]
68. Krummenacher, G.; Ong, C.S.; Koller, S.; Kobayashi, S.; Buhmann, J.M. Wheel defect detection with machine learning. *IEEE Trans. Intell. Transp. Syst.* **2017**, *19*, 1176–1187. [[CrossRef](#)]
69. Shi, D.; Šabanovič, E.; Rizzetto, L.; Skrickij, V.; Oliverio, R.; Kaviani, N.; Ye, Y.; Bureika, G.; Ricci, S.; Hecht, M. Deep learning based virtual point tracking for real-time target-less dynamic displacement measurement in railway applications. *Mech. Syst. Signal Process.* **2022**, *166*, 108482. [[CrossRef](#)]
70. Wu, Y.; Jin, W.; Li, Y.; Sun, Z.; Ren, J. Detecting Unexpected Faults of High-Speed Train Bogie Based on Bayesian Deep Learning. *IEEE Trans. Veh. Technol.* **2020**, *70*, 158–172. [[CrossRef](#)]
71. Xie, J.; Li, T.; Yang, Y.; Jin, W. Learning features from high speed train vibration signals with deep belief networks. In Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, 6–11 July 2014; pp. 2205–2210.
72. Xu, G.; Liu, M.; Jiang, Z.; Söfker, D.; Shen, W. Bearing fault diagnosis method based on deep convolutional neural network and random forest ensemble learning. *Sensors* **2019**, *19*, 1088. [[CrossRef](#)]
73. He, M.; He, D. Deep learning based approach for bearing fault diagnosis. *IEEE Trans. Ind. Appl.* **2017**, *53*, 3057–3065. [[CrossRef](#)]
74. Zou, Y.; Zhang, Y.; Mao, H. Fault diagnosis on the bearing of traction motor in high-speed trains based on deep learning. *Alex. Eng. J.* **2021**, *60*, 1209–1219. [[CrossRef](#)]
75. Zhan, Y.; Lin, K.; Zhan, H.; Guo, Y.; Sun, G. A unified framework for fault detection of freight train images under complex environment. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 1348–1352.
76. Sun, J.; Xiao, Z.; Xie, Y. Automatic multi-fault recognition in TFDS based on convolutional neural network. *Neurocomputing* **2017**, *222*, 127–136. [[CrossRef](#)]
77. Xiao, L.; Wu, B.; Hu, Y.; Liu, J. A hierarchical features-based model for freight train defect inspection. *IEEE Sens. J.* **2019**, *20*, 2671–2678. [[CrossRef](#)]
78. Ye, T.; Zhang, Z.; Zhang, X.; Chen, Y.; Zhou, F. Fault detection of railway freight cars mechanical components based on multi-feature fusion convolutional neural network. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 1789–1801. [[CrossRef](#)]
79. Zaman, A.; Ren, B.; Liu, X. Artificial intelligence-aided automated detection of railroad trespassing. *Transp. Res. Rec.* **2019**, *2673*, 25–37. [[CrossRef](#)]
80. Gao, H.; Huang, Y.; Li, H.; Zhang, Q. Multi-Sensor Fusion Perception System in Train. In Proceedings of the 2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS), Suzhou, China, 14–16 May 2021; pp. 1171–1176.
81. Wang, Z.; Wu, X.; Yu, G.; Li, M. Efficient rail area detection using convolutional neural network. *IEEE Access* **2018**, *6*, 77656–77664. [[CrossRef](#)]
82. Wang, Y.; Wang, L.; Hu, Y.H.; Qiu, J. RailNet: A segmentation network for railroad detection. *IEEE Access* **2019**, *7*, 143772–143779. [[CrossRef](#)]
83. Liu, H.; Liu, C.; He, S.; Chen, J. Short-Term Strong Wind Risk Prediction for High-Speed Railway. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4243–4255. [[CrossRef](#)]

84. Arvidsson, T.; Andersson, A.; Karoumi, R. Train running safety on non-ballasted bridges. *Int. J. Rail Transp.* **2019**, *7*, 1–22. [[CrossRef](#)]
85. Ding, Y.L.; Sun, P.; Wang, G.X.; Song, Y.S.; Wu, L.Y.; Yue, Q.; Li, A.Q. Early-warning method of train running safety of a high-speed railway bridge based on transverse vibration monitoring. *Shock Vib.* **2015**, *2015*, 518689. [[CrossRef](#)]
86. Choi, J.Y.; Kim, J.H.; Chung, J.S.; Lee, S.G. Evaluation of Train Running Safety for Direct Fixation Concrete Track on Light Rapid Transit. *J. Korean Soc. Saf.* **2017**, *32*, 41–46.
87. Jang, S.Y.; Yang, S.C. Assessment of train running safety, ride comfort and track serviceability at transition between floating slab track and conventional concrete track. *J. Korean Soc. Railw.* **2012**, *15*, 48–61. [[CrossRef](#)]
88. Lee, H.; Han, S.Y.; Park, K.; Lee, H.; Kwon, T. Real-Time Hybrid Deep Learning-Based Train Running Safety Prediction Framework of Railway Vehicle. *Machines* **2021**, *9*, 130. [[CrossRef](#)]
89. Heidarysafa, M.; Kowsari, K.; Barnes, L.; Brown, D. Analysis of railway accidents' narratives using deep learning. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; pp. 1446–1453.
90. Loy-Benitez, J.; Heo, S.; Yoo, C. Soft sensor validation for monitoring and resilient control of sequential subway indoor air quality through memory-gated recurrent neural networks-based autoencoders. *Control Eng. Pract.* **2020**, *97*, 104330. [[CrossRef](#)]
91. Huang, P.; Wen, C.; Fu, L.; Peng, Q.; Tang, Y. A deep learning approach for multi-attribute data: A study of train delay prediction in railway systems. *Inf. Sci.* **2020**, *516*, 234–253. [[CrossRef](#)]
92. Tang, Q.; Yang, M.; Yang, Y. ST-LSTM: A deep learning approach combined spatio-temporal features for short-term forecast in rail transit. *J. Adv. Transp.* **2019**, *2019*, 8392592. [[CrossRef](#)]
93. Zhang, D.; Peng, Y.; Zhang, Y.; Wu, D.; Wang, H.; Zhang, H. Train Time Delay Prediction for High-Speed Train Dispatching Based on Spatio-Temporal Graph Convolutional Network. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 2434–2444. [[CrossRef](#)]
94. Wolf, J.; Richter, R.; Döllner, J. Asset Detection in Railroad Environments using Deep Learning-based Scanline Analysis. In Proceedings of the VISAPP 2021: 16th International Conference on Computer Vision Theory and Applications, Virtual, 8–10 February 2021; pp. 465–470.