Hindawi
Journal of Advanced Transportation
Volume 2020, Article ID 8841810, 10 pages
https://doi.org/10.1155/2020/8841810

*Research Article*

# Neural Network-Based Train Identification in Railway Switches and Crossings Using Accelerometer Data

**Rostislav Krč**,[1] **Jan Podroužek**,[1] **Martina Kratochvílová**,[1] **Ivan Vukušič**,[2,3] **and Otto Plášek**[2]

[1] *Institute of Computer Aided Engineering and Computer Science, Faculty of Civil Engineering, Brno University of Technology, Brno 602 00, Czech Republic*
[2] *Institute of Railway Structures and Constructions, Faculty of Civil Engineering, Brno University of Technology, Brno 602 00, Czech Republic*
[3] *Výzkumný Ústav Železniční, a.s. (VUZ), Prague 142 00, Czech Republic*

Correspondence should be addressed to Rostislav Krč; rostislav.krc@vutbr.cz

Received 14 September 2020; Revised 23 October 2020; Accepted 12 November 2020; Published 24 November 2020

Academic Editor: Petr Dolezel

This paper aims to analyse possibilities of train type identification in railway switches and crossings (S&C) based on accelerometer data by using contemporary machine learning methods such as neural networks. That is a unique approach since trains have been only identified in a straight track. Accelerometer sensors placed around the S&C structure were the source of input data for subsequent models. Data from four S&C at different locations were considered and various neural network architectures evaluated. The research indicated the feasibility to identify trains in S&C using neural networks from accelerometer data. Models trained at one location are generally transferable to another location despite differences in geometrical parameters, substructure, and direction of passing trains. Other challenges include small dataset and speed variation of the trains that must be considered for accurate identification. Results are obtained using statistical bootstrapping and are presented in a form of confusion matrices.

## 1. Introduction

Railway switches and crossings (S&C) are important components of railway infrastructure. Dynamic effects of passing trains are higher than in case of a straight track and are affected by factors such as train speed, S&C geometry, fastening stiffness, and substructure material [1]. With increasing traffic and growing demands on the infrastructure, reliability and safety of S&C must be ensured. Large demands on maintenance occur especially on high-speed tracks [2]. Generally, three different maintenance approaches can be applied—corrective, preventive, and predictive [3].

Modern predictive approaches require real-time monitoring and data collection to evaluate S&C condition and apply appropriate countermeasures when needed [4]. Accelerometer or deflection sensors are simple and reliable devices that can be mounted directly in the S&C structure for monitoring the dynamic response. Gradual changes over time for the same train type and speed may indicate an emerging defect in S&C structure [5] and provide an early warning to the infrastructure operators. Therefore, train type must be recognized from the data to evaluate changes in S&C.

Project S-CODE (Switch and Crossing Optimal Design and Evaluation [6]) presented requirements for the next generation of S&C [7] and also introduced *next generation of control, monitoring, and sensing system* that, among others, will be able to determine the type of passing train based on accelerometer data. This system is referred to as Train Identification System (TIS). Recent studies also proposed to utilize machine learning techniques for predictive maintenance [8]. Train type was already successfully identified in a straight track [9]. Identification of trains in S&C is a more challenging task as more factors affecting sensor measurements must be considered.

Data can be obtained either from sensors mounted on trains or track. For successful train identification, it is important to recognize defects on trains such as flat wheels and not consider data from defected trains in S&C evaluation. Train defects such as wheel flats have been already detected by sensors mounted on trains [10] or track [11]. Critical samples containing defected wheels can be identified from the accelerometer signal by state-of-the-art pattern recognition techniques [12].

Machine learning methods can be used with benefit for processing a large amount of data. Methods such as support vector machines (SVMs) have already been incorporated for condition monitoring of railway infrastructure [13]. In this paper, neural networks are used for train identification as they are suitable for time series classification problems [14, 15]. Once trained, neural networks are also advantageous in terms of performance which may be useful for future in situ TIS.

The aim of this paper is to introduce possibilities of train type identification directly in S&C using neural networks and accelerometer data. This approach is unique and has not been attempted to date. Two locations and four S&C are considered, and several use case scenarios are presented in order to evaluate the transferability of machine learning models between different locations. Results for multiple train classes as well as various neural network architectures are discussed.

## 2. Data and Methods

*2.1. Data Acquisition.* Data used for train identification were obtained by in situ measurements from multiple accelerometer sensors placed in different positions around the common crossing of the S&C. The common crossing contained no movable parts. Therefore, passing trains caused increased acceleration impulses due to interruption of the rail continuity as wheels of the train hit the crossing nose. In a case of a movable crossing that is used in some S&C designs especially for high-speed tracks, these impulses would be lower but still detectable [16].

The full dataset contains signals from 6 single-axis accelerometers in $Z$-direction, 2 three-axis accelerometers in $X$, $Y$, and $Z$ directions, and 8 displacement sensors in $Z$-direction as shown in Figure 1. The sampling frequency of the sensors was 10 kHz. Sensors were placed either on ballast bed, sleeper, or directly on a rail near the crossing nose.

*2.2. Characteristics of Locomotive Classes.* Seven locomotive classes were chosen for identification as they vary in geometry, weight, or undercarriage stiffness: class 150/151 (denoted as 151), classes 162/163 (denoted as 163), class 362/363 (denoted as 363), class 380, Pendolino 680 (denoted as 680), Stadler 480 (denoted as 480), and class Siemens ES64U2/ES64U4 (denoted as Taurus). Geometrical parameters and weights for each class are shown in Table 1.

Data were obtained from two nearby locations on the same railway corridor in the Czech Republic: Choceň (referred to as Location 1) and Ústí nad Orlicí (referred to as Location 2). Two S&C were present in each location and their parameters differed between locations. Both S&C in Location 1 had different geometry (suitable for higher speeds), substructure parameters, and also an opposite direction of train passages compared to the S&C in Location 2. Another difference was that trains with locomotive class 363 had lower mean speed in Location 2 as they stopped in a nearby station. The speed of the trains was measured by a radar velocity gun with ±2 km/h accuracy. Measurements for each locomotive class and their speeds are listed in Table 2 for Location 1 and Table 3 for Location 2.

*2.3. Localization of the Locomotive Part.* Locomotive part of the accelerometer signal was used for the identification since locomotives are usually better maintained compared to the regular carriages. The variance of locomotive weights is also lower. Approaches used for locomotive localization from the whole signal were based on peak detection. Root mean square (RMS) value was calculated by equation (1) using a sliding window of size $d = n_1 - n_2$ for peak localization. Grouping of nearby peaks was done by mean shift clustering with bandwidth parameter $\alpha$:

$$\mathrm{RMS}_{n_2,n_1} = \sqrt{\frac{1}{n_2 - n_1} \sum_{n_1}^{n_2} |x(n)|^2}. \tag{1}$$

The size of the sliding window for RMS was chosen to $d = 0.02$ s. Peaks were then limited by a minimal amplitude value that was calculated dynamically using quantile of the whole signal between $q_{\mathrm{lim}} = 0.85 \sim 0.95$. Mean shift clustering with bandwidth parameter $\alpha = 0.03 \sim 0.033$ s distance was applied in order to group nearby peaks. All parameters were chosen empirically based on mean train speed. These methods served only for preprocessing of the given dataset and are not the aim of this research.

Each peak in an accelerometer signal represents an axle of a train and a two-peak group represents a bogie. Therefore, the signal can be divided into four-peak groups where the first group represents a locomotive which is followed by carriages as subsequent groups. This algorithm proved itself useful in data preprocessing and automatic extraction of the locomotive part of the signal as it was applied on a dataset which contained mostly signals with low levels of noise.

Example of an accelerometer signal generated by train with a locomotive of class 380 at speed 162 km/h passing through a S&C is shown in Figure 2. All axles of the train can be easily recognized as peaks in the signal. Detail of the locomotive part of this signal is shown in Figure 3.

*2.4. Methodology for Classification.* The high cost of corrective maintenance and risk of accidents require a robust solution for train type identification as it will be part of the S&C real-time monitoring system. The S-CODE project was proposed to incorporate accelerometer signals to determine the type of passing train [6]. Accelerometer sensors will be mounted in situ in the S&C structure and it is expected that a

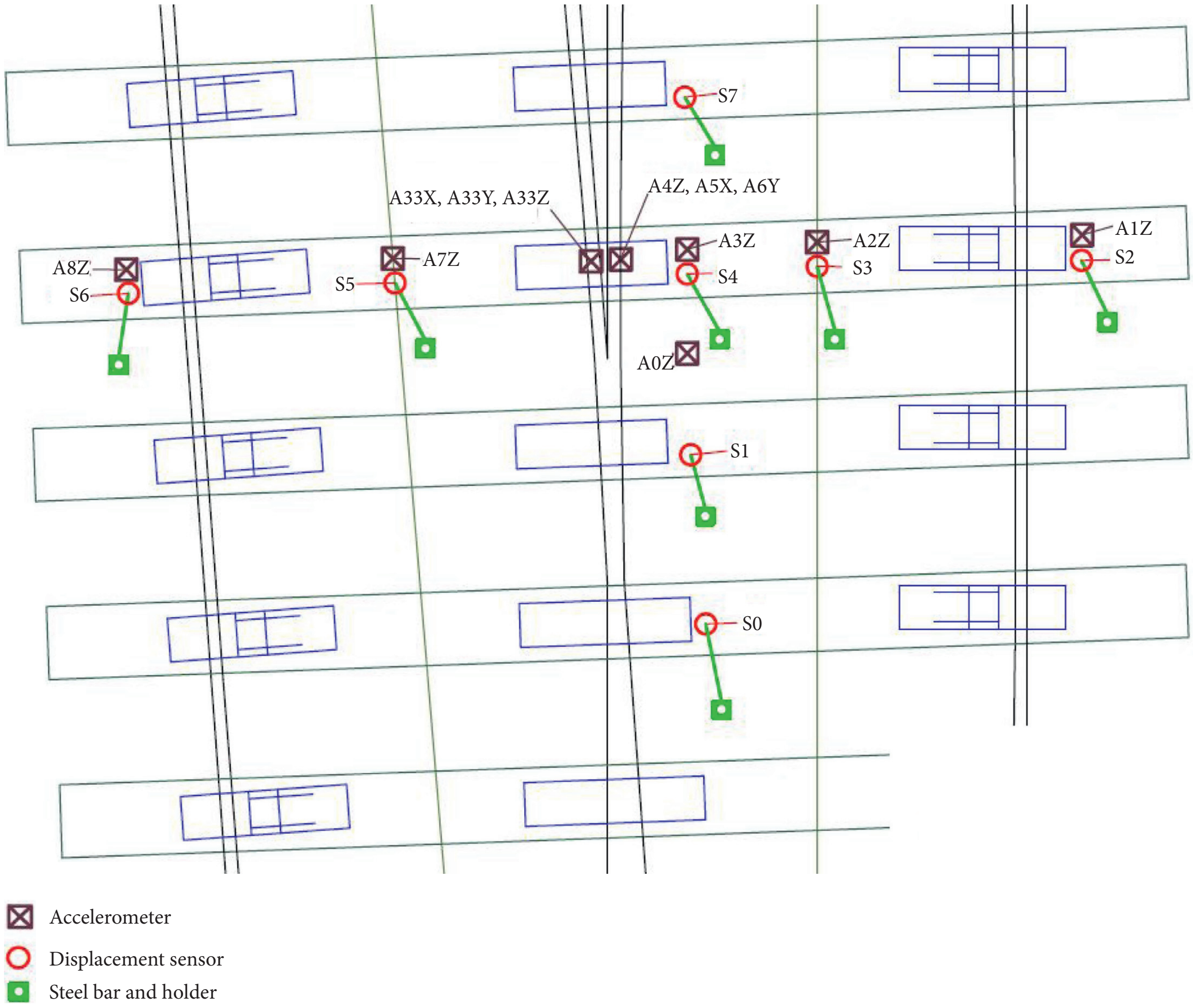

Figure 1: Sensor placement layout around the common crossing of the S&C used for data acquisition.

Table 1: Geometrical parameters and weights of different locomotive classes.

| Locomotive class | 151 | 163 | 363 | 380 | 480 | 680 | Taurus |
|---|---|---|---|---|---|---|---|
| Distance between pivots (m) | 8.3 | 8.3 | 8.3 | 8.7 | 16.0 | 19.0 | 9.9 |
| Axle spacing (m) | 3.2 | 3.2 | 3.2 | 2.5 | 2.7 | 2.7 | 3.0 |
| Weight (t) | 82.0 | 84.0 | 87.0 | 86.0 | 150.0[1] | 57.0 | 87.0 |

[1] Total weight of the whole five-car train.

Table 2: Location 1: measured locomotives and their speeds.

| Locomotive class | 151 | 163 | 363 | 380 | 480 | 680 | Taurus |
|---|---|---|---|---|---|---|---|
| Number of measurements (-) | 10 | 8 | 8 | 9 | 6 | 7 | 6 |
| Mean speed (km/h) | 133.2 | 106.5 | 129.6 | 147.4 | 159.3 | 154.4 | 145.3 |
| Speed standard deviation (km/h) | 15.5 | 35.5 | 13.0 | 13.0 | 4.4 | 4.7 | 9.8 |

large amount of data will be collected over time, so appropriate methods must be chosen for further data processing.

As stated in [13], machine learning methods, such as support vector machines (SVMs), are often used for monitoring and evaluation of the condition of railway infrastructure components [17] or for train defect detection from sensor data [11]. Using neural network-based models for time series classification is a common problem [18], and recent research mostly focuses on developing novel network

Table 3: Location 2: measured locomotives and their speeds.

| Locomotive class | 151 | 363 | 380 | 480 | 680 |
|---|---|---|---|---|---|
| Number of measurements (-) | 10 | 8 | 12 | 12 | 12 |
| Mean speed (km/h) | 122.0 | 91.9 | 128.1 | 147.0 | 128.5 |
| Speed standard deviation (km/h) | 5.7 | 14.8 | 4.9 | 12.7 | 4.3 |

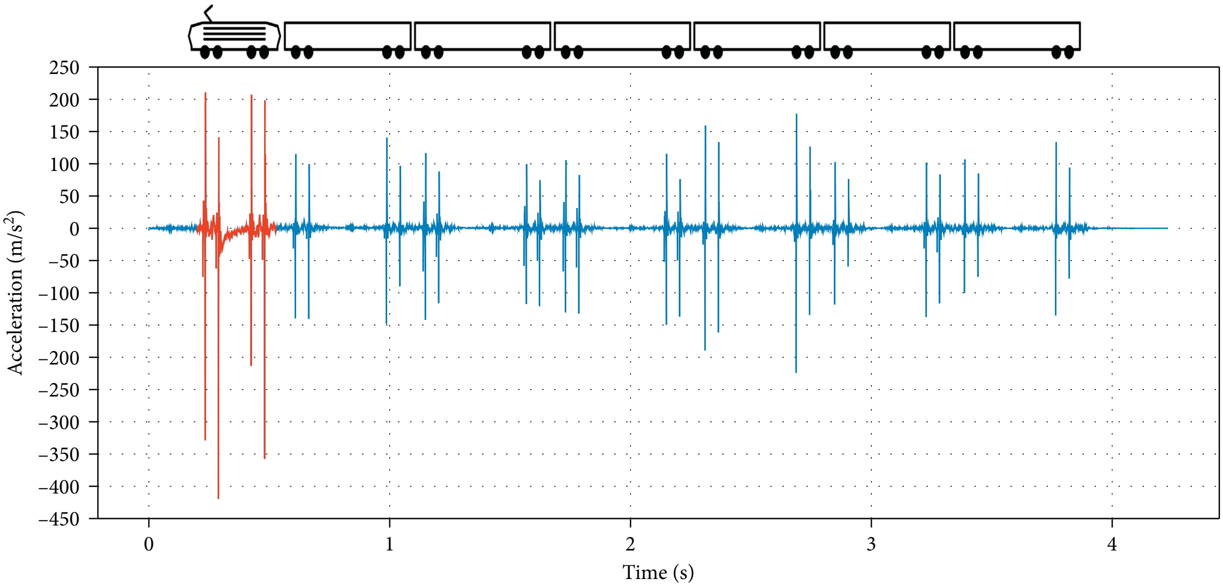

Figure 2: Accelerometer signal ($Z$-axis) during train passage over a crossing. The locomotive part is highlighted in red and shown in detail in Figure 3.

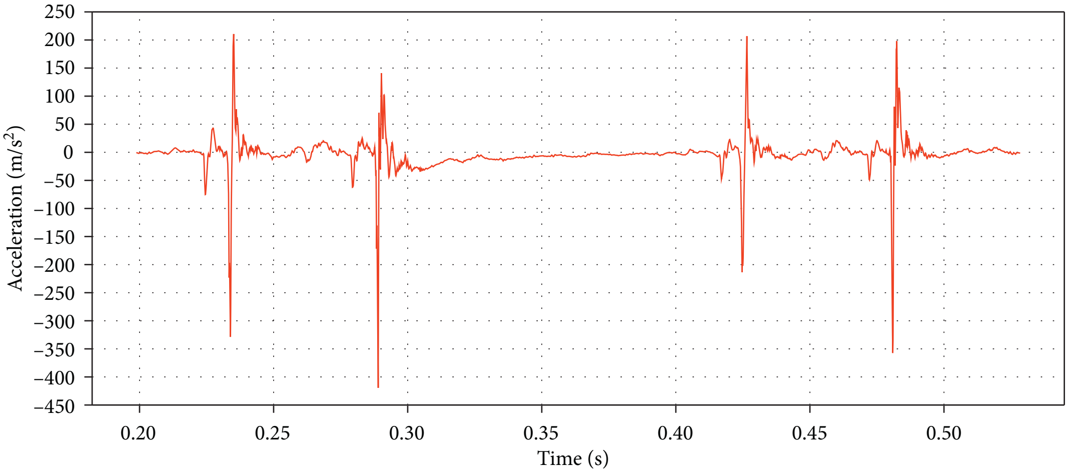

Figure 3: Detail of the locomotive part of the signal shown in Figure 2.

architectures such as modifications of residual neural network (ResNet) [19]. Convolutional neural networks are often used for the classification of time series data with an outstanding performance [20, 21] and are also widely used for the classification of accelerometer data and human activity recognition [14, 22]. In railway engineering, deep neural networks were successfully applied in areas such as fault diagnosis on trains [23] or for rail degradation prediction [24].

Given the high complexity of the train type identification problem in S&C, multiple neural network architectures will be examined in this paper in order to find an optimal design.

*2.5. Neural Network Design and Training.* Six different neural network architectures were evaluated for locomotive classification. Four multilayer perceptrons with either one

(MLP1), two (MLP2), or three (MLP3a, MLP3b) fully connected hidden layers. Hidden layer size was set to 100 neurons in all cases except MLP3b where 500 neurons were used. All perceptron models employed rectified linear activation function (ReLU) between layers except the output layer where softmax activation was applied. Using the softmax activation function in the output layer is a common practice [25] which has an advantage that the vector of output probabilities sums to 1.

A convolutional neural network (CNN) consisted of a convolutional layer with 64 filters of length 5 followed by a max-pooling layer of length 5 and a hidden layer of size 100. ReLU was used as an activation function between layers and softmax activation at the output.

The sixth and final architecture was a long short-term memory recurrent neural network (LSTM) with one LSTM layer with 50 hidden states followed by a fully connected layer with output softmax activation function.

The input size for all models was set to 1000, and the output size was the number of classified train types (i.e., 5 and 7). The training was done in 12 epochs and data were forwarded through the model in batches of size 4. The learning rate of the models was fixed to 0.001. Adam optimizer was selected for automatic differentiation [26], and mean squared error was used as a loss function. The number of trainable parameters and the number of layers for different neural network architectures are presented in Table 4.

*2.6. Normalization of Input Accelerometer Signals.* Specific features from the data can be selected as input to the neural networks to decrease complexity and improve training times. However, a whole accelerometer signal may be used without a need for extensive and domain-specific preprocessing. This approach also removes bias due to manually selected features [18] and improves performance, especially for in situ device.

In the first step, signals were normalized in both $X$- and $Y$-axes to prevent locomotive misclassification for different train speeds. The number of samples in available locomotive signals spanned between $1 \cdot 10^3$ and $1 \cdot 10^4$ depending on the sampling frequency of the sensors, train speed, and locomotive geometry. In the $X$-axis, signals were resampled to the input size which was chosen to 1000. This number of samples is sufficient as it preserves enough information with a lower number of samples than in the original signal (see Figure 4). In the $Y$-axis signals were normalized between values $-1$ and 1.

*2.7. Use Case Scenarios.* Four accelerometer channels A0Z, A2Z, A3Z, and A7Z were selected for train identification as they were similar in terms of phase shift and noise. Sensors A2Z, A3Z, and A7Z were placed on a sleeper under the crossing nose and sensor A0Z was placed in a ballast bed nearby as shown in Figure 1. These four channels were used separately in order to augment data and increase its variability as the sensors can generally be placed in arbitrary position around the crossing nose. The full dataset contained 108 train measurements from Location 1 and Location 2 giving in a total of 432 samples. To evaluate classification models for a different variety of data, these two locations were considered both independently and together, using only locomotive classes present in both locations (5 classes).

Four use case scenarios were considered as shown in Table 5. In scenarios A and B, the models were trained on all the samples from Location 1 and Location 2, respectively. In scenario C, the data from these two locations were combined. Size of the dataset remained relatively small despite using four accelerometer channels independently. Therefore, the bootstrapping technique [27] was utilized for scenarios A, B, and C in order to produce statistically relevant results. 10 models were trained and tested for each neural network architecture and each scenario, and the results were averaged to evaluate the overall performance of the given architecture [28]. For every model, the scenario dataset was shuffled and split in the way that at least two locomotive passages (i.e., 8 samples) for each class were available for testing.

Finally, the use case scenario D used data from Location 2 for training and the data from Location 1 for testing. This scenario aimed to evaluate a situation when the model for train identification is trained on the currently available data and then applied to another S&C.

## 3. Results

Substantial differences of classification accuracy between the use case scenarios, locomotive classes, and neural network architectures were observed due to factors such as the variance of train speeds, undercarriage geometry, or dynamic response of S&C structure. Despite these factors, the accuracy of the presented models is still relatively high compared to random classification.

Baseline accuracy (random classification) for scenario A is 14.3% and for scenarios B to D is 20.0% and is given by the inverse of the number of classified classes. Mean model accuracy for different scenarios spanned between 52.3% and 80.6% and is presented in Table 6 and Figure 5. The difference in the mean accuracy in the considered two locations (scenarios A and B) was 28.3% and has to be addressed to the higher data variability in Location 1 as more locomotive classes were classified and also the train speeds were more variable. Training models on data from one location and testing on the other (scenario D) resulted in a mean accuracy of 55.0%. Combining data from both locations together (scenario C) exhibited a mean accuracy of 72.9%. Confusion matrices were used for the evaluation of results.

Differences can also be observed between different neural network architectures (Table 6 and Figure 5). The flexibility of models varies as the number of trainable parameters differs (see Table 4). CNN shows the best accuracy in all scenarios compared to the other models since the convolutional layer enhances the ability of feature recognition in time series data. This architecture also contains the largest number of trainable parameters. All multilayer perceptrons (models MLP1, MLP2, MLP3a, and MLP3b) have only low variance in accuracy and with slightly decreasing trend for deeper architectures. Relatively poor mean accuracy was observed in LSTM due to difficulties in

Table 4: Number of layers and number of trainable parameters for the evaluated neural network architectures.

| Model | MLP1 | MLP2 | MLP3a | MLP3b | CNN | LSTM |
|---|---|---|---|---|---|---|
| Number of layers | 2 | 3 | 4 | 4 | 4 | 2 |
| Number of trainable parameters | 100 605 | 110 705 | 120 805 | 1 000 005 | 1 280 989 | 260 605 |

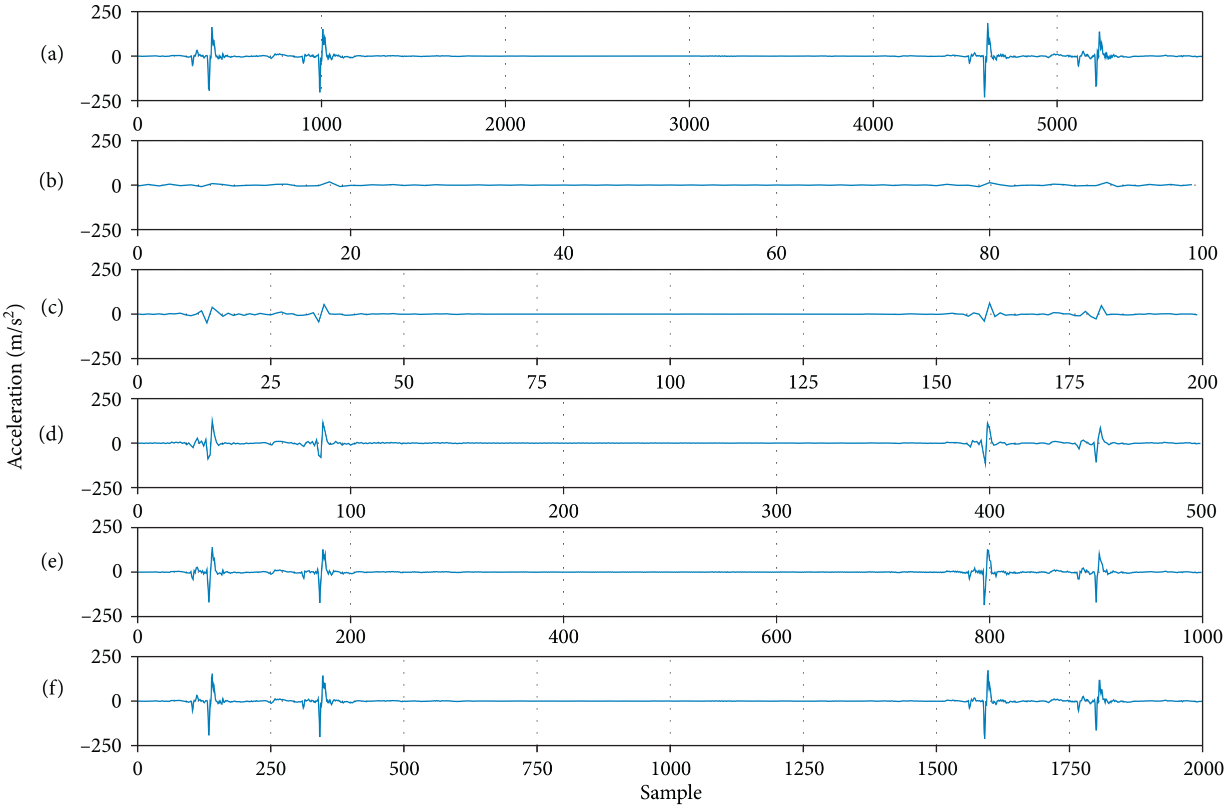

Figure 4: Effect of signal resampling to a different number of samples. (a) Original signal with 5791 samples. (b) Resampled to 100 samples. (c) Resampled to 200 samples. (d) Resampled to 500 samples. (e) Resampled to 1000 samples. (f) Resampled to 2000 samples.

Table 5: Setup of evaluated use case scenarios.

| Scenario | Location | Bootstrapping (no. of repeats) | No. of classes | Dataset size | Training size | Testing size |
|---|---|---|---|---|---|---|
| A | Location 1 | Yes (30) | 7 | 216 | 152 | 64 |
| B | Location 2 | Yes (30) | 5 | 216 | 164 | 52 |
| C | Location 1 and 2—mixed | Yes (30) | 5 | 376 | 308 | 68 |
| D | Location 2 (training), Location 1 (testing) | No | 5 | 376 | 216 | 160 |

Table 6: Mean accuracy (with standard deviation if applicable) for different neural network models and different scenarios. Baseline (random) accuracy is denoted as "Base." Visualization of this table is shown in Figure 5.

| Model/Scenario | Base (%) | Mean (%) | MLP1 (%) | MLP2 (%) | MLP3a (%) | MLP3b (%) | CNN (%) | LSTM (%) |
|---|---|---|---|---|---|---|---|---|
| A | 14.3 | 52.3 ± 7.9 | 50.9 ± 4.4 | 52.3 ± 8.2 | 51.4 ± 7.0 | 49.5 ± 8.2 | **60.0 ± 6.3** | 49.7 ± 7.2 |
| B | 20.0 | 80.6 ± 12.0 | 82.9 ± 5.7 | 87.3 ± 7.0 | 83.3 ± 6.0 | 81.2 ± 6.0 | **89.2 ± 6.9** | 59.8 ± 9.7 |
| C | 20.0 | 72.9 ± 9.9 | 76.2 ± 7.1 | 74.1 ± 5.2 | 73.7 ± 9.2 | 73.5 ± 4.9 | **80.6 ± 6.9** | 59.3 ± 9.8 |
| D | 20.0 | 55.0 | 57.5 | 58.8 | 53.7 | 53.1 | **72.5** | 34.4 |

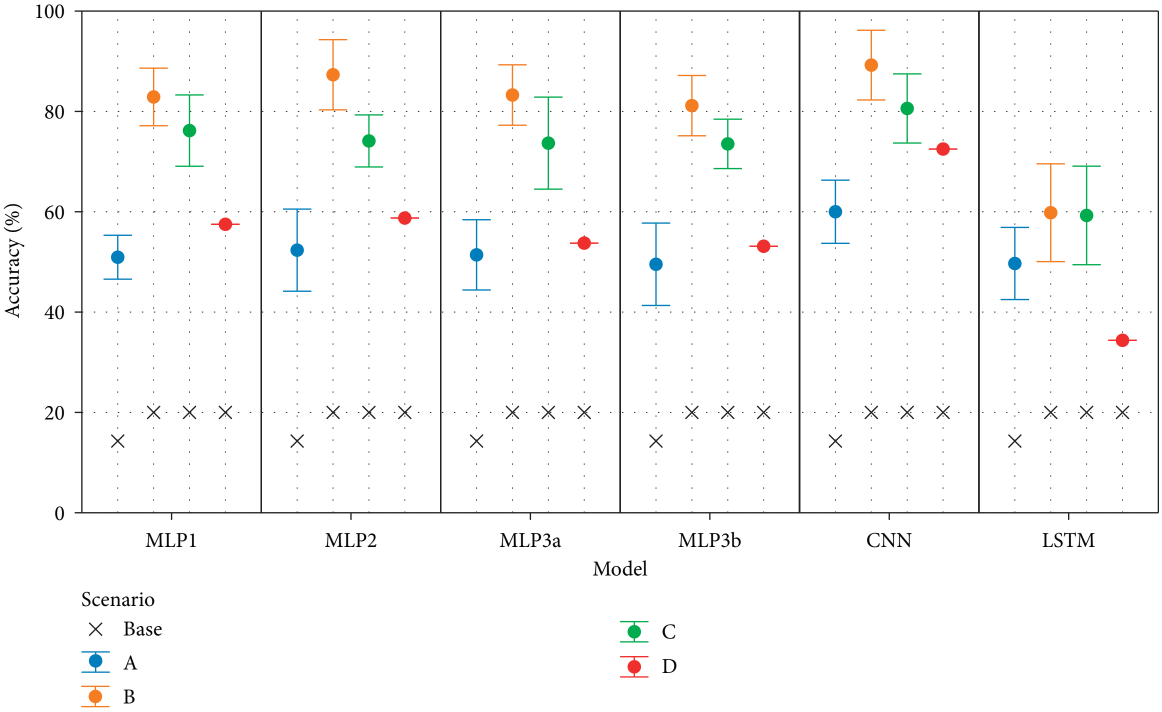

Figure 5: Mean model accuracies with standard deviation (if applicable) for different use case scenarios. Baseline (random) accuracy is denoted as “Base” and is marked in black.

the training process. Mean confusion matrices for the most accurate CNN architecture are presented in Figures 6–9.

Locomotive classes were also classified with varying accuracy. Pendolino 680, Stadler 480, and class 380 were identified with the highest mean accuracy due to their specific undercarriage geometry. On the contrary, mean accuracies in scenario A for the three mutually geometrically similar classes 151, 163, and 363 were lower. Differences in the classification accuracy for the same locomotive classes in different scenarios are to be addressed to the variance of speed. An overview of the mean accuracy of classification for each locomotive class is shown in Table 7.

## 4. Discussion

Results showed differences in accuracy for different scenarios, locomotives, and machine learning models which can be addressed to factors such as complex dynamic interaction of the train and S&C structure, multiple locomotive classes, similarities in locomotive undercarriage geometries, speed variance, and a relatively small amount of training data. The test scenario C that used data from one location for training and the other location for testing presented that neural network-based classifiers are generally transferable to S&C in different locations. Nevertheless, the model performance has to be improved by using a larger training dataset and more advanced architectures of the neural networks. Additionally, high uncertainty in case of trains with high-speed variance requires partitioning trains with different speeds into separate classes.

The highest classification accuracy of CNN was expected since it is the most commonly used architecture for this type of problem [18]. On the other hand, the lowest accuracy of LSTM compared to the other evaluated models may be attributed to the long input sequence as this architecture is generally suitable for time series classification [29]. Adding a convolutional layer to LSTM may also increase its accuracy as this architecture was successfully applied in a number of time series classification or prediction problems [30, 31].

Trains with different undercarriage geometry were identified with the highest accuracy contrary to the trains with similar geometry that were often mutually misclassified. Large speed variability should also be addressed for the poor classification accuracy for class 363.

It is expected that more accelerometer data from train passages through S&C will be available in the future. Advanced network architectures such as LSTM with convolutional layers [30] or ResNet [19] will be examined as well as more refined optimization of hyperparameters. Also, data augmentation techniques can be employed to increase dataset size and variability [32]. Another possible solution is to use transfer learning [22] and utilize a large amount of data available in other industries. Here, machine learning models can be trained on similar time series data and then fine-tuned for the locomotive classification problem.

The ultimate goal is to develop a full-featured solution for locomotive identification in order to evaluate changes in the dynamic response of S&C for the same train types and speeds as well as to detect defected trains and exclude them from the dataset to improve classification accuracy.

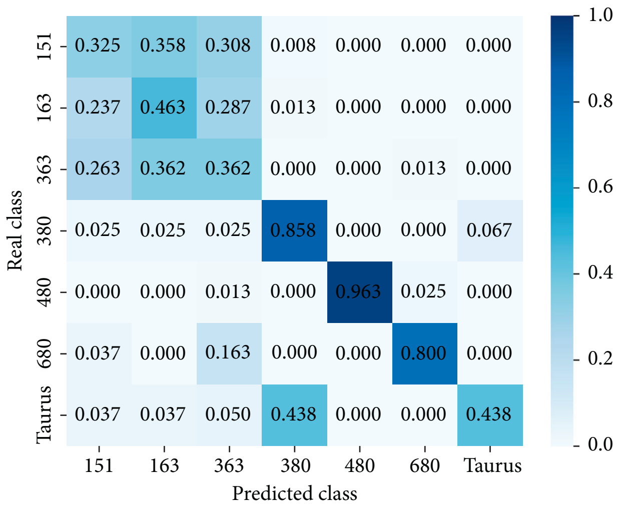

Figure 6: Mean confusion matrix for the best performing CNN for scenario A.

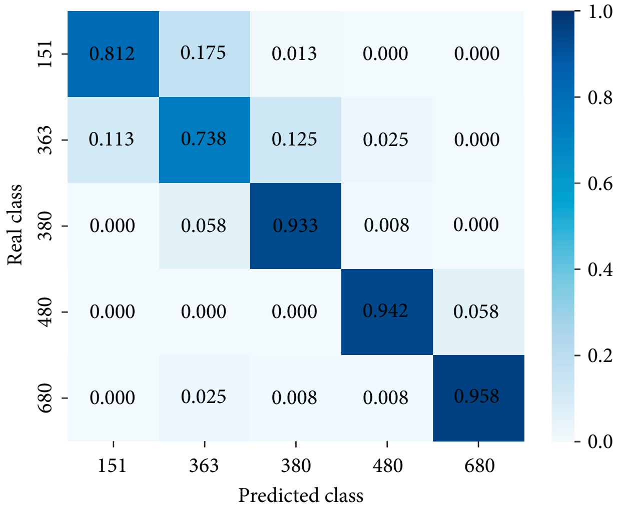

Figure 7: Mean confusion matrix for the best performing CNN for scenario B.

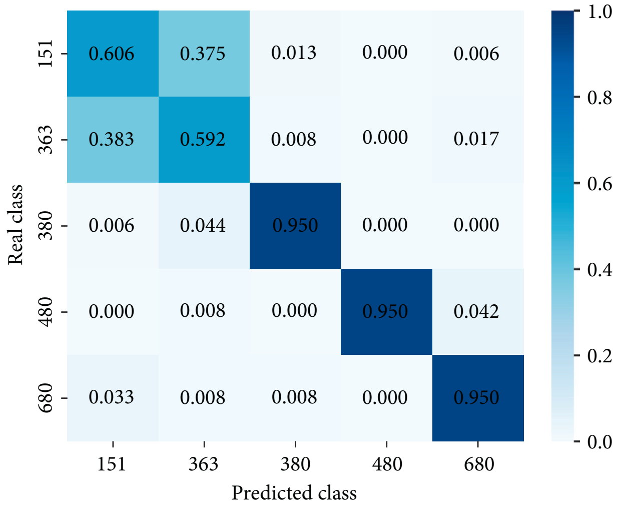

Figure 8: Mean confusion matrix for the best performing CNN for scenario C.

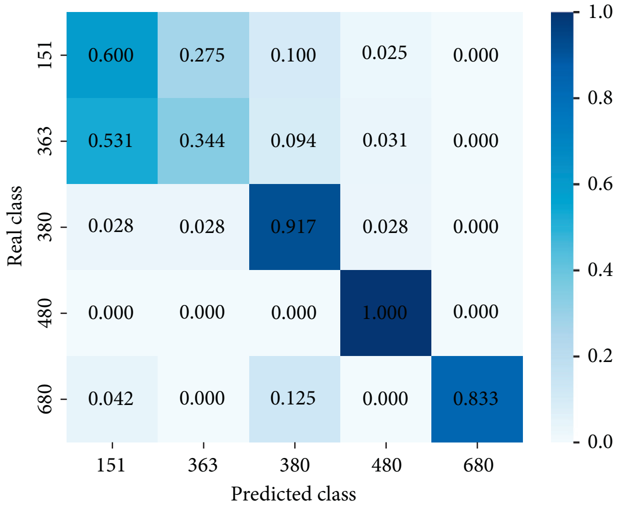

Figure 9: Confusion matrix for the best performing CNN for scenario D.

Table 7: Mean accuracy for different locomotive classes.

| Locomotive class/scenario | 151 (%) | 163 (%) | 363 (%) | 380 (%) | 480 (%) | 680 (%) | Taurus (%) |
|---|---|---|---|---|---|---|---|
| A | 30.1 | 37.5 | 38.8 | 61.0 | 87.9 | 73.8 | 44.0 |
| B | 81.5 | — | 47.7 | 88.7 | 82.2 | 92.2 | — |
| C | 59.3 | — | 46.3 | 87.7 | 81.3 | 89.6 | — |
| D | 50.4 | — | 17.2 | 58.8 | 81.0 | 77.1 | — |

## 5. Conclusions

Train identification based on accelerometer data in S&C using different neural network architectures was presented in this paper. The most important findings can be summarized as follows:

(i) Train type identification in S&C is feasible despite the increased complexity of the problem compared to a straight track.

(ii) Transferability of machine learning models between different locations is also possible. Models can be trained on data from one location and then applied to another, previously unseen location, with relatively high classification accuracy in spite of differences in S&C parameters. However, both locations evaluated in this paper are positioned on one railway corridor. It is therefore desirable to further verify the transferability of models between unrelated locations.

(iii) Accelerometer signals can be classified without a need for manual feature selection with respect to the limited computational capacity of the in situ device.

To enhance the robustness of evaluated models, only the locomotive part of the signal was used as locomotives are less variable in terms of weight and wheel geometry. However, locomotives with largely different speeds were incorrectly classified despite normalization of input data. Grouping of locomotives into speed categories is required in order to improve classification accuracy. Additionally, defected trains must be identified in advance and excluded from the dataset for successful train identification and subsequent evaluation of the dynamic response of S&C.

Comparison of four use case scenarios and six neural network architectures showed higher model performance for data with lower variability and vice versa. The best performing convolutional neural network proved to be a suitable baseline architecture for the locomotive classification problem. In further research, more advanced neural network architectures, as well as hyperparameter optimization, will be investigated.

## Data Availability

The data used in this work were provided exclusively by Správa železnic, the national railway infrastructure manager of the Czech Republic. Data can be provided on demand at the e-mail address info@spravazeleznic.cz.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

Financial support provided by the Technology Agency of the Czech Republic under the projects *Turnout 4.0* (CK01000091) and *Efficient spacetime predictions using machine learning methods* (TJ04000232) as well as the support of the project *Smart sensoric system for railways* (FAST/FSI-J-20-6265) is greatly acknowledged.

## References

[1] I. Vukušič, D. Vukušičová, K. Zaplatílek, J. Podroužek, J. Apeltauer, and M. Kratochvílová, "Dynamic effects diagnosis in railway switches and crossings within the S-code project," *Scientific and Technical Proceedings of Správa Železnic*, vol. 1, pp. 1–26, 2019.

[2] C. Zhang, Y. Gao, W. Li, L. Yang, and Z. Gao, "Robust train scheduling problem with optimized maintenance planning on high-speed railway corridors: the China case," *Journal of Advanced Transportation*, vol. 2018, Article ID 6157192, 16 pages, 2018.

[3] B. Dhillon, *Engineering Maintenance: A Modern Approach*, CRC Press, Boca Raton, Florida, USA, 2002.

[4] S. Huang, F. Zhang, R. Yu, W. Chen, F. Hu, and D. Dong, "Turnout fault diagnosis through dynamic time warping and signal normalization," *Journal of Advanced Transportation*, vol. 2017, Article ID 3192967, 8 pages, 2017.

[5] M. Sysyn, U. Gerber, O. Nabochenko, Y. Li, and V. Kovalchuk, "Indicators for common crossing structural health monitoring with track-side inertial measurements," *Acta Polytechnica (Prague, Czech Republic : 1992)*, vol. 59, no. 2, pp. 170–181, 2019.

[6] S-Code, http://www.s-code.info/about.

[7] O. Plasek, L. Raif, I. Vukusic, V. Salajka, and J. Zelenka, "Design of new generation of switches and crossings," in *Proceedings of the Conference on Future Trends in Civil Engineering 2019*, vol. 1, pp. 277–301, 2019.

[8] Z. Allah Bukhsh, A. Saeed, I. Stipanovic, and A. G. Doree, "Predictive maintenance using tree-based classification techniques: a case of railway switches," *Transportation Research Part C: Emerging Technologies*, vol. 101, pp. 35–54, 2019.

[9] E. Berlin and K. Van Laerhoven, "Sensor networks for railway monitoring: detecting trains from their distributed vibration footprints," in *Proceedings of the 2013 IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, Cambridge, MA, USA, May 2013.

[10] B. Liang, S. D. Iwnicki, Y. Zhao, and D. Crosbee, "Railway wheel-flat and rail surface defect modelling and analysis by time-frequency techniques," *Vehicle System Dynamics*, vol. 51, no. 9, pp. 1403–1421, 2013.

[11] G. Krummenacher, C. S. Ong, S. Koller, S. Kobayashi, and J. M. Buhmann, "Wheel defect detection with machine learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1176–1187, 2018.

[12] J. Podrouzek, C. Bucher, and G. Deodatis, "Identification of critical samples of stochastic processes towards feasible structural reliability applications," *Structural Safety*, vol. 47, pp. 39–47, 2014.

[13] M. Hamadache, S. Dutta, O. Olaby, R. Ambur, E. Stewart, and R. Dixon, "On the fault detection and diagnosis of railway switch and crossing systems: an overview," *Applied Sciences*, vol. 9, no. 23, p. 5129, 2019.

[14] A. Ignatov, "Real-time human activity recognition from accelerometer data using convolutional neural networks," *Applied Soft Computing*, vol. 62, pp. 915–922, 2018.

[15] S. Vernekar, S. Nair, D. Vijaysenan, and R. Ranjan, "A novel approach for classification of normal/abnormal phonocardiogram recordings using temporal signal analysis and machine learning," in *Proceedings of the 2016 Computing in Cardiology Conference*, Vancouver, Canada, September 2016.

[16] M. Z. Hamarat, S. Kaewunruen, and M. Papaelias, "Contact conditions over turnout crossing noses," *IOP Conference Series Materials Science and Engineering*, vol. 471, no. 6, pp. 1–12, 2019.

[17] H. Tsunashima, "Condition monitoring of railway tracks from car-body vibration using a machine learning technique," *Applied Sciences*, vol. 9, no. 13, 2019.

[18] H. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[19] J. Wu, Z. Zhang, Y. Ji, S. Li, and L. Lin, "A ResNet with GA-based structure optimization for robust time series classification," in *Proceedings of the 2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering*, Hangzhou, China, April 2019.

[20] C.-L. Liu, W.-H. Hsaio, and Y.-C. Tu, "Time series classification with multivariate convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4788–4797, 2019.

[21] B. Qian, Y. Xiao, Z. Zheng et al., "Dynamic multi-scale convolutional neural network for time series classification," *IEEE Access*, vol. 8, p. 1, 2020.

[22] Recent Research From Swiss Federal Institute Of Technology Highlight Findings In Convolutional Neural Networks (Real-Time Human Activity Recognition From Accelerometer Data Using Convolutional Neural Networks), (Report), Journal of Engineering, 2018.

[23] H. Hu, B. Tang, X. Gong, W. Wei, and H. Wang, "Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 2106–2116, 2017.

[24] A. Falamarzi, S. Moridpour, M. Nazem, and R. Hesami, "Rail degradation prediction models for tram system: Melbourne case study," *Journal of Advanced Transportation*, vol. 2018, Article ID 6340504, 8 pages, 2018.

[25] I. Goodfellow, *Deep Learning*, MIT Press, Cambridge, MA, USA, 2016.

[26] D. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2017, https://arxiv.org/abs/1412.6980.

[27] R. W. Johnson, "An introduction to the bootstrap," *Teaching Statistics*, vol. 23, no. 2, pp. 49–54, 2001.

[28] Y. Xu and R. Goodacre, "On splitting training and validation set: a comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning," *Journal of Analysis and Testing*, vol. 2, no. 3, pp. 249–262, 2018.

[29] Z. Lipton, D. Kale, and R. Wetzel, "Phenotyping of clinical time series with LSTM recurrent neural networks," 2017, https://arxiv.org/abs/1510.07641.

[30] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "LSTM fully convolutional networks for time series classification," *IEEE Access*, vol. 6, pp. 1662–1669, 2018.

[31] C.-J. Huang, "A deep CNN-LSTM model for particulate matter (PM2.5) forecasting in Smart cities," *Sensors*, vol. 18, no. 7, p. 2220, 2018.

[32] G. Forestier, J. Weber, and P.-A. Muller, "Data augmentation using synthetic data for time series classification with deep residual networks," 2018, https://arxiv.org/abs/1808.02455.