

Automated Inspection System with GPS and Deep Learning in Urban Rail Safety and Efficiency

Ljubomir Vračar¹, Dragan Marinković^{2,3}, Milan Stojanović¹, Miloš Milovančević^{*4}

¹ University of Niš, Faculty of Electronic Engineering, A. Medvedeva 14, 18000 Niš, Serbia, email: ljubomir.vracar@elfak.ni.ac.rs, milan.stojanovic@elfak.ni.ac.rs

² Technische Universität Berlin, Department of Structural Mechanics and Analysis, Strasse des 17. Juni 135, 10623 Berlin, Germany, e-mail: dragan.marinkovic@tu-berlin.de

³ Institute of Mechanical Science, Vilnius Gediminas Technical University, 10105 Vilnius, Lithuania

⁴ University of Niš, Faculty of Mechanical Engineering, A. Medvedeva 14, 18000 Niš, Serbia, e-mail: dragan.marinkovic@masfak.ni.ac.rs, milos.milovancevic@masfak.ni.ac.rs

* Corresponding author

Abstract: Paper focuses on a novel automated rail inspection system, incorporating advanced technologies such as GPS for precise location tracking and GSM/GPRS modules for efficient data transmission. The system uses deep learning to analyze vibration data collected during train transitions, enabling predictive maintenance and enhancing rail safety within urban smart city frameworks. This system represents a significant step forward in intelligent transportation systems by automating and improving the efficiency and accuracy of rail inspections. The use of deep learning for data analysis underscores the potential of AI in infrastructure maintenance, potentially setting a new standard for rail safety protocols. The paper discusses the technical design of the sensor node, the integration of GPS and GSM/GPRS modules, the application of deep learning algorithms, and the analysis of the system's performance through testing and validation. The implications of such a system for smart city infrastructure and urban planning, as well as potential future enhancements and applications of the technology.

Keywords: SMART sensors; AI; vibration; maintenance

1 Introduction

The projections, which are based on the rapid advancement of industrialization and the urbanization of human civilization, indicate that by the year 2050, 85 percent of the world's population would be living in urban regions. In a world where the population is always growing, these regions are responsible for most of the appropriate resource use and environmental problems that occur. A growing number of people all around the globe are becoming interested in the idea of a smart city, also known as an intelligent city, digital city, e-city, or smart community, as it is described in many pieces of literature. Using cutting-edge technology, the notion of a smart city may facilitate the provision of solutions to these problems as well as suitable services for the purpose of enhancing the quality of life in urban areas.

Infrastructure that is both intelligent and sustainable, which is accomplished via the use of modern technical solutions, is one of the primary features of a smart city. Providing a living space that is pleasant, economically efficient, environmentally friendly, and convenient is the fundamental goal of a smart city. This is accomplished via the use of real-time technology that automate certain chores. Using monitoring systems and communication technologies, the notion of a smart city represents a more effective method for optimizing the utilization of resources and other services. One of the most important aspects of smart cities is the use of cutting-edge technology that may enhance and automate activities inside metropolitan areas, therefore, making these environments greener and more sustainable. There has been a significant influence on the development of these systems brought about by the fast growth of digitalization and communication technologies, as well as sensors and actuators [1-3].

The implementation of integrated sensor systems is the first stage in the process of developing smart cities. Sensors are devices that can detect and measure physical characteristics, as well as give real-time data that is required for observation, control, and decision making. Because of this, sensors are crucial components for building smart cities [4-6]. The structure that is known as a sensor node is created by a single sensor or many sensors, together with a power source and a device for data transmission. A network, also known as a wireless sensor network (WSN), is formed by all sensor nodes that have a central node (sink) to which they send data. The signals that are obtained by sensors can be interpreted by humans or integrated electronic circuits that process acquired data. This eliminates the need for manual inspection, which in turn reduces the cost and energy consumption. Additionally, it provides an autonomous control and monitoring of various parts in smart city structures.

A crucial component of smart cities is the implementation of intelligent transportation and mobility systems, which are based on Intelligent Transportation Systems (ITS). The objectives of these systems is to improve energy conservation and emission reduction [7], improve efficiency of decision making algorithms [8], detect energy losses [9], improve safety [10], etc. Several mobile and permanent

sensors are included into the Intelligent Transport System (ITS) to provide adaptive vehicle navigation, intelligent fleet management and traffic monitoring, detection of road incidents, avoidance of congestion, and other similar functions. Environmental factors, such as temperature and humidity, chemical parameters, such as oxidation and corrosion, and mechanical characteristics, such as strain, stress, and deformation, are used to describe the conditions of all traffic infrastructures [11]. The visual examination that is carried out manually is often insufficient since it is possible that irregularities will still go unnoticed during the process. In the field of intelligent transportation systems (ITS), sensors are used in public infrastructures like roads, bridges, and railroads to offer awareness that allow for more effective utilization of resources and minimize the need for human inspection.

Deep Learning, which is a technique that is based on Artificial Intelligence and Block Chain, is being implemented to increase the accuracy, efficacy, and intelligence of various smart city management systems [12]. To increase the accuracy and real-time performance of abnormal behavior detection smart video monitoring, the intelligent video technology, which is based on convolutional neural network (CNN) deep learning has been developed [13].

Deep learning is used in the Intelligent Transportation System (ITS) for several areas of traffic management and the monitoring of traffic structures. The precise forecast of passenger flow is of enormous value in the operation and administration of subways. It enables operators to create decision support systems for subway operations that are more accurate, which is particularly crucial for major events [14]. The application of Artificial Neural Networks (ANNs) in decision support systems allows for the estimation of the hourly passenger populations at bus stops, as well as the formulation of certain suggestions for the management of bus lines [15]. The reduction of the number of accidents that occur on the roads is an essential component of traffic, and this may be accomplished using predictions derived from CNN [16]. It is possible for faults in the fasteners to be responsible for the vertical displacement of the rail and vibrations that occur during the transition of the train. This is because of the friction that occurs between the train wheel and the rail. Manual rail control has become obsolete because of the proliferation of high-speed rail lines. This is since it is time-consuming, requires the expertise of specialists, and inspections are only performed when the line is empty. This procedure may be replaced by the visual method for locating faults in rail fasteners by using a hybrid approach that is based on deep learning [17].

This paper describes the sensor node equipped by accelerator adequate for automated inspection of the rail by measuring the vibrations during the train transition. System is integrated with GPS module to determine the position of the train (place of measurement on the rail), and GSM/GPRS modules to send the real-time data. The obtained data are analyzed using the deep learning technology which provide the prediction about the future state of the rail. The paper is organized as follow: Section 2 describes the hardware of sensor node in detail. Section 3 shows

the applied deep learning technology used for analyzing the obtained data. Section 4 presents the results followed by the conclusion and ideas for further research.

2 Hardware Design of the Sensor Node

The used device was developed around Microchip microcontroller PIC18F46K22 [17]. This microcontroller was chosen because of well-known architecture and stable hardware characteristics. Simplified block diagram of the sensor node are shown on Figure 1.

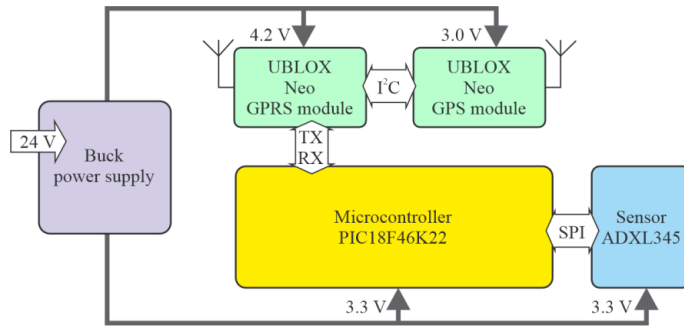


Figure 1

Simplified block diagram of the sensor node

The power supply of the sensor node is 24 V, what makes it suitable for industrial standard and application. This voltage is reduced to 4.2 V for supplying the GSM/GPRS module using the buck power supply, a further adjusted to 3.3 V for the microcontroller and the sensor, and 3.0 V for GPS module's power supply, using the regulators shown in Figure 2.

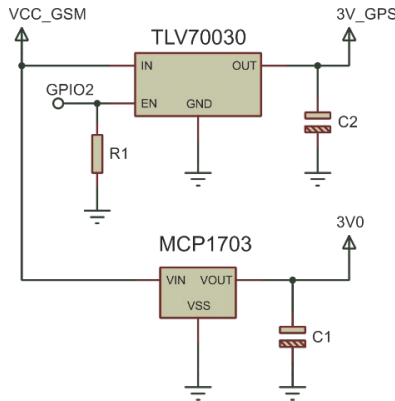


Figure 2

The power supply regulators for microcontroller, sensor and GPS module power supply

For GSM/GPRS module, Ublox LEON module was chosen for the same reason. Microcontroller relates to GSM/GPRS module through its first UART. Communication between microcontroller and GSM module is established via AT commands. Figure 3 and Figure 4 show part of circuit with microcontroller and GSM/GPRS module.

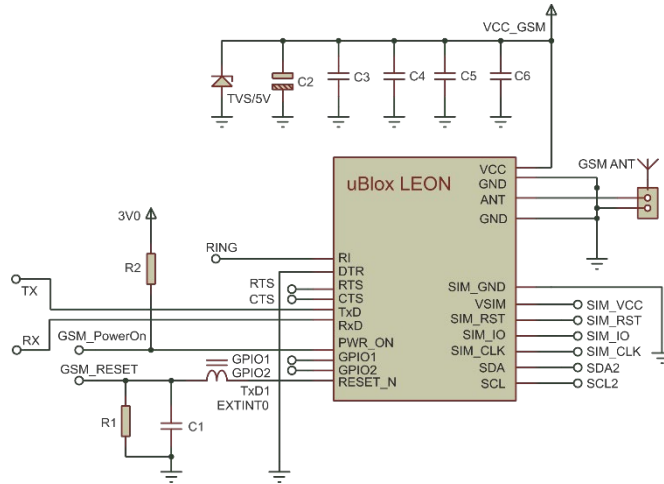


Figure 3

The electronic circuit of the GSM/GPRS module

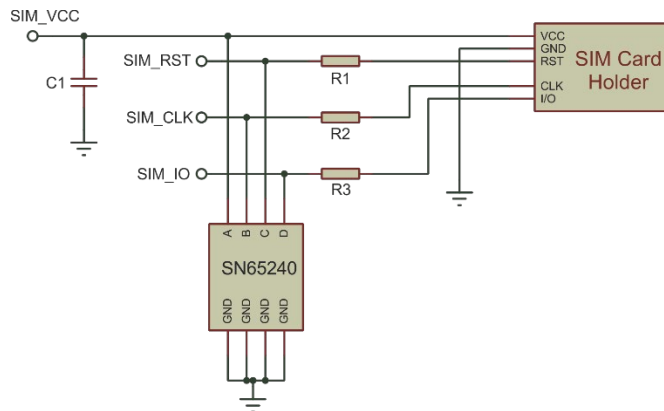


Figure 4

The power supply regulators for microcontroller, sensor and GPS module power supply

The GPS module, which electronic circuit is shown in Figure 5. Ublox Neo6 with low power consumption was selected. GPS module communicates with GSM/GPRS module through I2C interface.

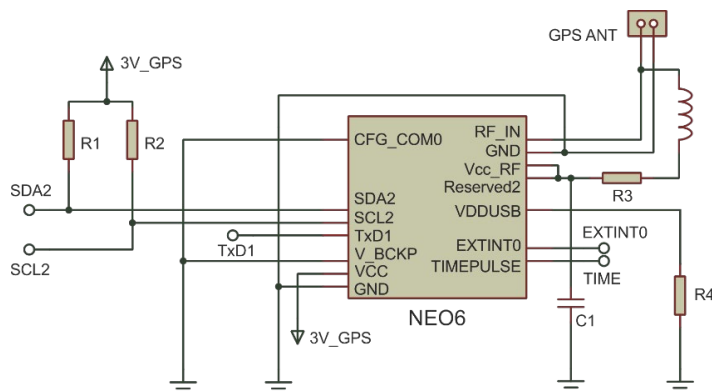


Figure 5

The electronic circuit of the GPS module

Vibrations are measured using the three axes digital accelerometer ADXL345 (Figure 6). This sensor has a low-power consumption, and provides accurate measurement. Digital output data are accessible through the I²C or SPI interface that was chosen because of higher data transmission rate.

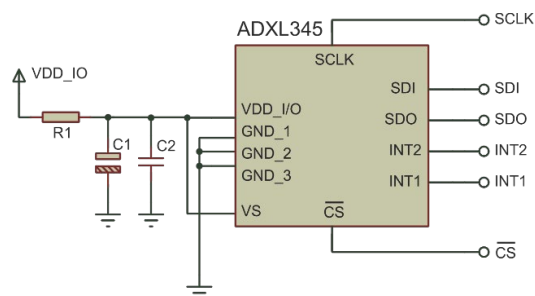


Figure 6

The electronic circuit of ADXL345 accelerometer

The measured value of acceleration and coordinates are transmitted to the server through implementing the Message Queuing Telemetry Transport (MQTT) protocol. MQTT is a Client Server (Broker) publish/subscribe protocol in application layer of TCP/IP. Server accepts messages from clients or send messages to clients. There are two types of clients, publishers and subscribers. The publisher sends message on a specific topic to the server. Following the server's receipt of the published message from the publisher, the payload of the message is sent to each client that has subscribed to that particular subject. As can be observed, the MQTT protocol was developed for asynchronous communication, which is characterized by the simultaneous occurrence of subscriptions or publishing to or from many entities. While other protocols, such as HTTP, have a much larger footprint, the MQTT protocol has a far lower footprint. As was said before, this makes MQTT significantly more suited for situations that have limited resources. Despite the fact

that the MQTT protocol offers a multitude of benefits, not all brokers that are based on MQTT possess the capabilities to authenticate entities or encrypt data [18]. On the other hand, size of packet in MQTT protocol allows to send previously encrypted data, which size is 64 bits [19]. In that case, broker forward encrypted message to subscriber which perform decryption of the message.

The complete sensor node realized on printed circuit board is shown in Figure 7.

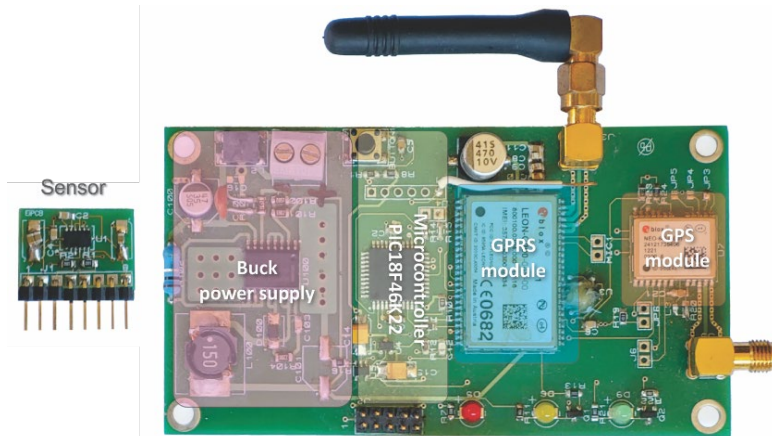


Figure 7

Top view of the printed circuit board of the sensor node

The Adaptive Neuro-Fuzzy Inference System (ANFIS) [20] was used to evaluate how time-lagged variables affect urban railway track lateral and vertical acceleration predictions. The learning and reasoning capabilities of artificial neural networks and fuzzy logic make ANFIS ideal for nonlinear and complicated time-series data. This research used a basic measuring technique that measured accelerations [21, 22, 23].

Both lateral and vertical accelerations were recorded on a test track under real-world situations. Measurements were taken in both directions along the track to obtain data from all relevant track sections to study. This method identified dynamic behaviours and the effects of temporal fluctuations on accelerations, allowing ANFIS to accurately predict acceleration patterns from time-lagged data.

This measuring technology was used with ANFIS to create a strong prediction model to improve track dynamics knowledge and urban railway system maintenance plans [24, 25, 26].

3 Methodology and Materials

3.1 Measuring Methodology

Running safety and ride quality were assessed using the EN 14363 standard, which offers a thorough framework for assessing railway cars under diverse operating scenarios. The technique in EN 14363 parts 5.2.2.1 and 5.2.2.2 emphasises acceleration measurements to simplify evaluation. This technique precisely captures essential train dynamic behaviours, simplifying conventional testing. According to Article 5.4.5 of EN 14363, all relevant test track sections were assessed in both travel directions to account for directional performance implications. A diesel-powered motor train with two passenger motor units linked by an inter-car clutch and bypass system was used for the test. Each motor unit has two two-axle bogies: a steering bogie for driving and a clutch bogie for non-steering. The train also has four turntables for stability and orientation. The train carried 18 passengers during testing. Based on intercity travel rules, the train weighed 89,103 kg, assuming 80 kg each passenger, including baggage. The train wheels have UIC-ORE-compliant rolling surfaces. The train had travelled 1500 km before testing. Therefore, the wheel rolling surfaces showed minimal wear, which was considered during analysis. In Figure 8, the experimental setup was created to correctly quantify running safety and driving quality. This arrangement permitted precision acceleration data gathering across track conditions and revealed train operating dynamics under common passenger loads and track layouts [27].





Figure 8
Experimental measuring setup

3.2 ANFIS Methodology

The fuzzy inference system underpins the architecture of the Adaptive Neuro-Fuzzy Inference System (ANFIS). Layer 1 receives input variables. These inputs are transformed into fuzzy values. Membership functions, mathematical constructions that convert each point in the input space to a membership value between 0 and 1, simplify this translation.

In this work, the membership function is critical for system performance, particularly with nonlinear data. The study uses bell-shaped membership functions. The bell-shaped curve of these functions is smooth and continuous. This approach was made because the bell-shaped function excels in nonlinear data regression.

I like the bell-shaped membership function for various reasons. First, its smooth and flexible structure enables for more nuanced and accurate fuzzy set representation, which is vital for capturing real-world data variability and uncertainty. Second, the function's parameters may be readily altered to help the ANFIS network properly respond to incoming data and improve regression.

The bell-shaped membership function helps the ANFIS network simulate complicated, nonlinear input-output interactions. The ANFIS network is useful for regression analysis and prediction in many disciplines since it can handle nonlinear data.

Bell-shaped membership function is defined as follows:

$$\mu(x) = \text{bell}(x; a_i, b_i, c_i) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i} \right)^2 \right]^{b_i}} \quad (1)$$

where $\{a_i, b_i, c_i\}$ is the parameters set and x is input.

In the first layer, fuzzy inputs are processed and multiplied in the second layer to determine the rule firing strength. The third layer normalizes the outputs from the second layer, ensuring consistency across rule activations. In the fourth layer, the system applies the inference mechanism, where the fuzzy rules are evaluated and the results are transformed into crisp values. Finally, the output layer aggregates the signals from the previous layers and produces the final crisp output, representing the system's decision or prediction.

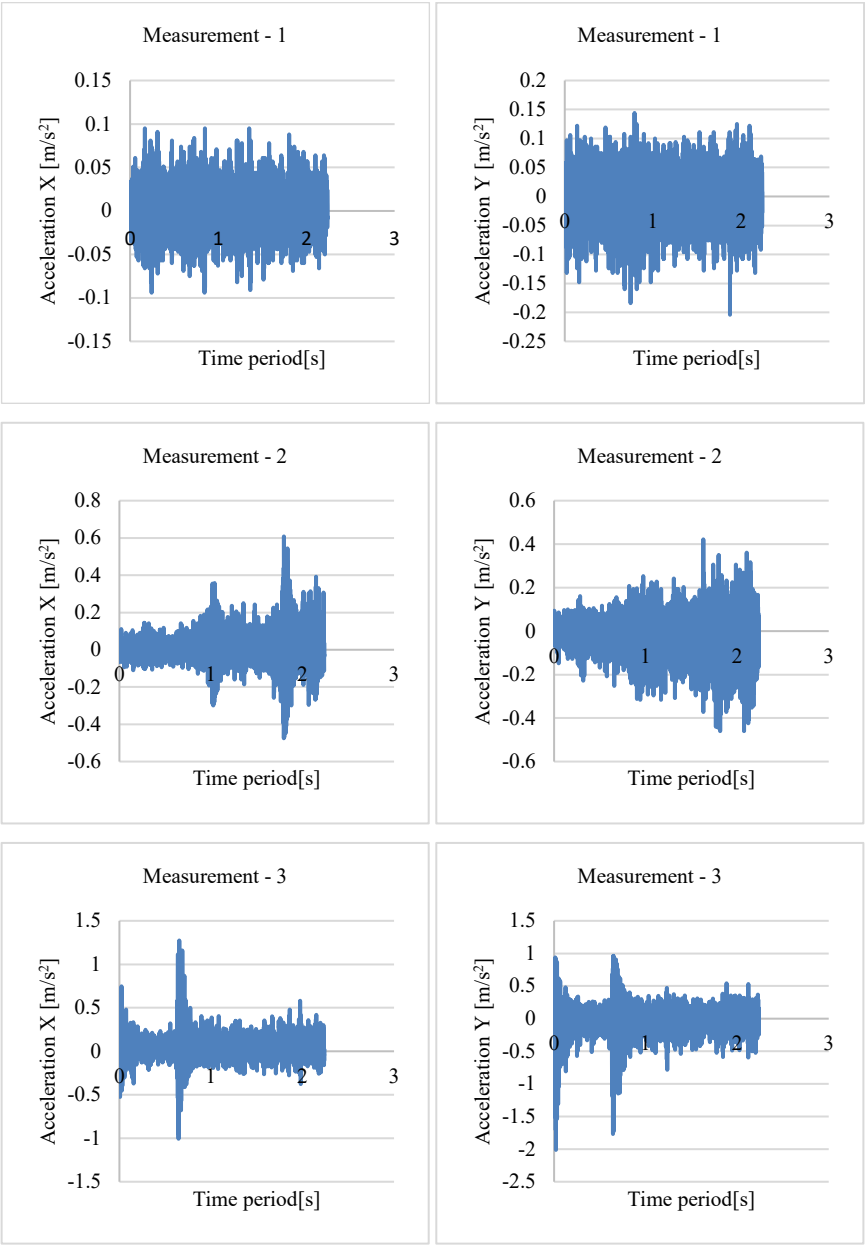
4 Results

4.1 Experimental Data Measurements

In the study, the lateral and vertical accelerations were measured to assess the height of the axes of the two outer axle assemblies on the frames of two rotating stands—one drive and one free bogie per turntable—and the car body at floor level above the central bolts of one bogie's wheels. These measurements were obtained using acceleration data. Additionally, motorcycles with toilets, designated as car 1, were equipped with measurement instruments, aided by a pulse encoder that tracked both distance and speed. A total of six acceleration values were considered for analysis (Figure 9). The pulse encoder was mounted on the shaft assembly housing to monitor speed and distance, which were used as control quantities.

The measured signals were filtered in accordance with point 7 of the testing program to generate the assessment parameters. These parameters comprised 16 in total: six for safety evaluation and ten for dynamic driving performance. The analysis divided the test sections into segments of 250 m, 100 m, and 70 m, depending on the test area and program specifics. Statistical processing was performed on these segments, either on N or $2N$ members of the statistical set, where N represents the number of sections within a particular curve region.

The value of each parameter for a given test section was determined by the cumulative curve of the observed parameter distribution. This distribution was based on specific percentages of frequency of occurrence, which allowed the generation of a histogram of parameter distribution by test area. These processed parameters were then used to evaluate the running safety and dynamic driving style of the vehicle during the tests.



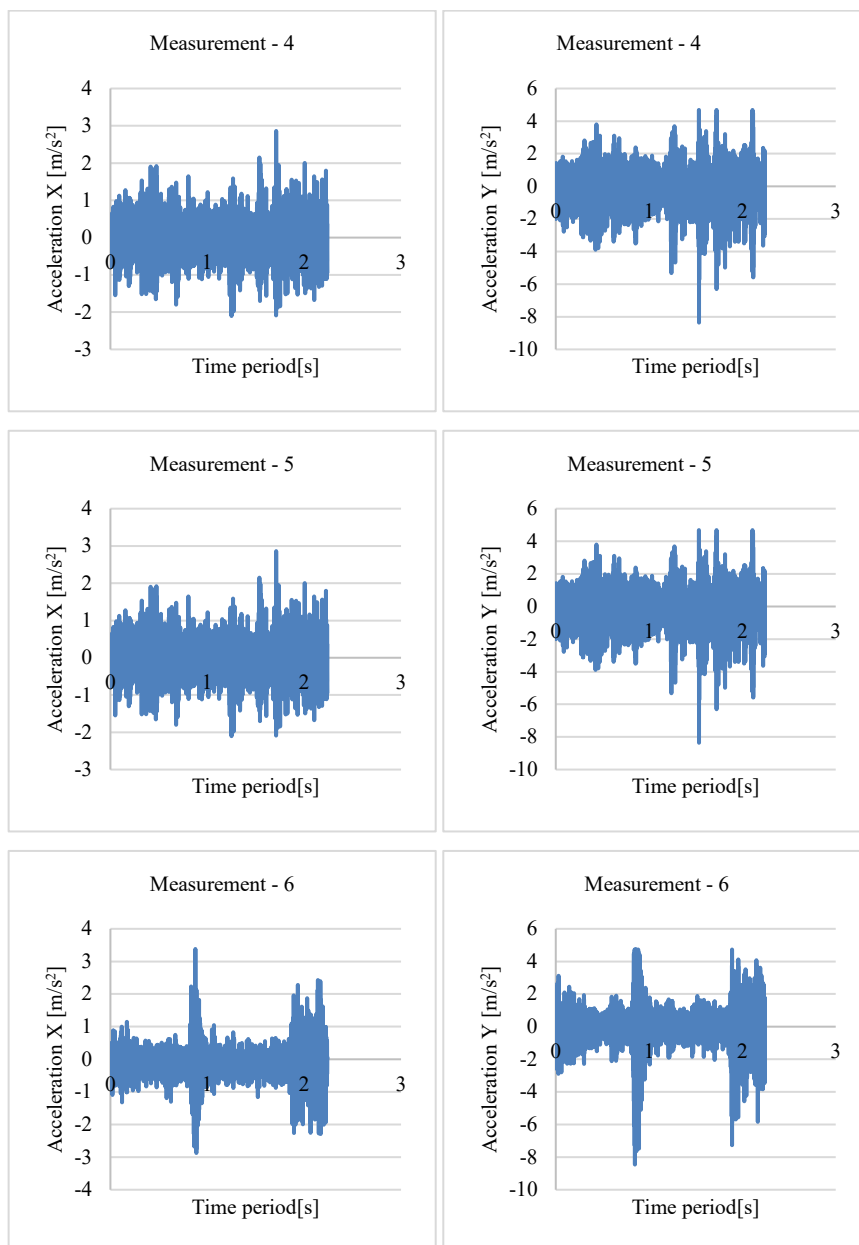


Figure 9
Lateral (X) and vertical (Y) acceleration

4.2 ANFIS Data Clustering

The ANFIS methodology was employed to perform data clustering based on the prediction of acceleration using time-lagged data. The time-lagged data was generated through the following process:

$$x(t-18), x(t-12), x(t-6), x(t).$$

Thus, the procedure involves four input variables, and the output is predicted as $x(t+6)$. The selection of relevant inputs is crucial for clustering, as it allows the removal of inputs with minimal influence on the prediction accuracy. The dataset was divided into two subsets: a training set consisting of odd-indexed samples, and a checking set consisting of even-indexed samples. Six data samples were used, corresponding to six measurements in total.

Four distinct ANFIS models were created, each using one input selected from the four possible candidates. The results for each of the six measurements were obtained based on this approach, providing insights into the predictive accuracy and clustering effectiveness for the given time-lagged acceleration data.

The results of the ANFIS-based prediction models for time-lagged data are presented across six measurements, where both training (trn) and checking (chk) errors are analyzed for each lag. For each measurement, four different time-lagged inputs ($x(t-18)x(t-18)x(t-18)$, $x(t-12)x(t-12)x(t-12)$, $x(t-6)x(t-6)x(t-6)$, and $x(t)x(t)x(t)$) were used to predict the output $x(t+6)x(t+6)x(t+6)$. Similarly, the same analysis was performed for the lateral acceleration (X) and vertical acceleration (Y) data, with the following observations:

Measurement 1 (X Data)

- For $x(t-18)x(t-18)x(t-18)$, the training error is 0.0252 and the checking error is 0.0242.
- For $x(t-12)x(t-12)x(t-12)$, the training error remains the same at 0.0252, but the checking error slightly increases to 0.0243.
- The training and checking errors for $x(t-6)x(t-6)x(t-6)$ are 0.0250 and 0.0245, respectively.
- $x(t)x(t)x(t)$ shows a higher training error of 0.0255 and a checking error of 0.0246.

Measurement 2 (X Data)

- There is a noticeable increase in errors compared to Measurement 1.
- $x(t-18)x(t-18)x(t-18)$ shows a training error of 0.0703 and a checking error of 0.1108.
- Errors decrease as the lag decreases, with $x(t)x(t)x(t)$ achieving the lowest training (0.0538) and checking (0.0958) errors.

Measurement 3 (*X* Data)

- All lags show relatively high errors.
- $x(t-18)x(t-18)x(t-18)$ has a training error of 0.1615 and a checking error of 0.1130, while the other lags yield similar results with minimal variations in training and checking errors.

Measurement 4 (*X* Data)

- This measurement shows the highest overall errors.
- $x(t-18)x(t-18)x(t-18)$ has a training error of 0.4920 and a checking error of 0.5612.
- The errors for other lags follow a similar trend, with the $x(t)x(t)x(t)$ input yielding a slightly lower training error (0.4784) but maintaining a similar checking error.

Measurement 1 (*Y* Data)

- The *Y* data (vertical acceleration) shows smaller errors compared to the *X* data in Measurement 1.
- $y(t-18)y(t-18)y(t-18)$ yields a training error of 0.0437 and a checking error of 0.0416, which remain consistent across all lags, indicating stable predictions.

Measurement 2 (*Y* Data)

- Similar to *X* data, the errors increase for the *Y* data in Measurement 2.
- $y(t-18)y(t-18)y(t-18)$ shows a training error of 0.0766 and a checking error of 0.1169, with errors decreasing slightly as the lag decreases, reaching 0.0725 for training and 0.1208 for checking.

Measurement 3 (*Y* Data)

- Errors are significantly higher in this measurement.
- $y(t-18)y(t-18)y(t-18)$ has a training error of 0.2221 and a checking error of 0.1606.
- The errors for other lags are comparable, showing minimal variations.

Measurement 4 (*Y* Data)

- This measurement exhibits the highest errors for *Y* data, similar to *X* data.
- $y(t-18)y(t-18)y(t-18)$ yields a training error of 1.0376 and a checking error of 1.3411.
- Other lags show slightly lower training errors, but the checking errors remain high, with $y(t)y(t)y(t)$ resulting in 1.0398 for training and 1.3304 for checking.

Measurement 5 (X Data)

- Errors for this measurement follow a similar pattern to Measurement 4.
- $x(t_{-18})x(t_{-18})x(t_{-18})$ shows a training error of 0.4920 and a checking error of 0.5612, with minor variations across the other lags.

Measurement 6 (X Data)

- The errors decrease slightly compared to previous measurements.
- $x(t_{-18})x(t_{-18})x(t_{-18})$ yields a training error of 0.4599 and a checking error of 0.4817.
- The errors across other lags are comparable, with minimal variations.

Measurement 5 (Y Data)

- Y data shows consistent high errors for this measurement.
- $y(t_{-18})y(t_{-18})y(t_{-18})$ has a training error of 1.0376 and a checking error of 1.3411, with the other lags following a similar trend.

Measurement 6 (Y Data)

- The errors in Measurement 6 are slightly lower than in Measurement 5.
- $y(t_{-18})y(t_{-18})y(t_{-18})$ yields a training error of 1.1411 and a checking error of 1.0627, with $y(t)y(t)y(t)$ achieving the lowest errors (1.1542 for training and 1.0162 for checking).

the results across all six measurements demonstrate varying levels of prediction accuracy. Lower errors are observed for the initial measurements, while the later ones exhibit higher discrepancies between training and checking sets, particularly in Y data for Measurements 4 through 6. The differences in performance across time lags and between lateral and vertical acceleration data suggest that the choice of time-lagged inputs plays a critical role in the accuracy of the ANFIS model predictions.

The analysis of the results after grouping the most influential inputs reveals distinct patterns in the predictive significance of each input. The findings are as follows:

- $X(t_{-18})$ and $Y(t_{-18})$ have no influence on the future predictions, as indicated by a score of 0.
- $X(t_{-12})$ is selected once, while $Y(t_{-12})$ is selected three times, indicating that $Y(t_{-12})$ plays a more significant role in the prediction of $Y(t_{+6})$.
- $X(t_{-6})$ is selected twice, whereas $Y(t_{-6})$ is selected once, suggesting moderate relevance of $X(t_{-6})$ in predicting future $X(t_{+6})$ values.
- $X(t)$ is the most frequently selected input for the prediction of $X(t_{+6})$, with a frequency of three, indicating that the current value $X(t)$ has the strongest influence on future acceleration predictions in the X direction.

- Similarly, $Y(t)$ is selected once, but $Y(t_{-12})$ remains the most frequently chosen input for predicting future $Y(t_{+6})$, highlighting the importance of a prior time step for vertical acceleration predictions.

In conclusion, $X(t)$ emerges as the most influential input for the prediction of future $X(t_{+6})$ values, while $Y(t_{-12})$ plays a critical role in predicting future $Y(t_{+6})$ values. The inputs from the earliest time step $X(t_{-18})$ and $Y(t_{-18})$ are determined to have no significant influence on the predictions, suggesting that more recent data points are more relevant for accurate forecasting. This demonstrates the effectiveness of clustering in isolating the most impactful inputs for ANFIS-based time-series predictions.

Conclusions

The integration of GPS and GSM/GPRS modules is highlighted for its capability to enable real-time data transmission, significantly enhancing the system's efficiency in providing timely and accurate assessments of rail conditions. The deployment of deep learning technology for data analysis is emphasized as a key factor in augmenting the system's predictive capabilities, thus fostering a proactive approach to rail maintenance and safety.

The system's innovative ability to measure vibrations during train transitions and analyze this data in real-time is underscored as a considerable advancement in intelligent transport systems (ITS) within the context of smart city frameworks. By automating the inspection process and leveraging advanced data analysis techniques, the system presents a more efficient, accurate, and cost-effective solution in contrast to traditional manual inspection methods.

A focus on further refining the deep learning models to improve predictive accuracy and exploring the integration of additional sensor types. This would provide a more comprehensive assessment of rail conditions and expand the system's application to other critical infrastructure components within the smart city ecosystem. Addressing potential challenges related to data security and privacy, system scalability, and interoperability with existing urban infrastructure management systems is also identified as crucial for the widespread adoption of this technology.

The development of this sensor node and its application in automated rail inspection highlights the significant potential of integrating advanced technologies into smart city infrastructure. This integration is poised to improve urban living conditions, enhance public safety, and contribute to the sustainable development of urban areas, marking a substantial step forward in the evolution of smart cities.

References

- [1] Baranovski I., Stankovski S., Ostojić G., Nikolić V., Simonović M., Stanojević M. (2022) Toward a smart ecosystem with automated services. Facta Universitatis, Series: Mechanical Engineering, doi: 10.22190/FUME220525033B

- [2] Alharbi N., Soh B. (2019) Roles and Challenges of Network Sensors in Smart Cities. IOP Conference Series: Earth and Environmental Science, 322, <https://iopscience.iop.org/article/10.1088/1755-1315/322/1/012002>
- [3] Bara M., Vitković N., Stanković Z., Rajić M., Turudija, R. (2024) Description and Utilization of an Educational Platform for Clean Production in Mechanical Engineering. Spectrum of Mechanical Engineering and Operational Research, 1(1), 145-158, <https://doi.org/10.31181/smeor11202413>
- [4] S. Shamsir, I. Mahbub, S. K. Islam, A. Rahman, (2017). Applications of sensing technology for smart cities. 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), Boston, MA, USA, pp. 1150-1153, doi: 10.1109/MWSCAS.2017.8053132
- [5] Ramírez-Moreno MA, Keshtkar S, Padilla-Reyes DA, Ramos-López E, García-Martínez M, Hernández-Luna MC, Mogro AE, Mahlknecht J, Huertas JI, Peimbert-García RE, et al. (2021) Sensors for Sustainable Smart Cities: A Review. Applied Sciences, 11(17), 8198, <https://doi.org/10.3390/app11178198>
- [6] Enlund D., Harrison K., Ringdahl R., Börütecene A., Löwgren J., Angelakis V. (2022) The role of sensors in the production of smart city spaces. Big Data & Society, 9 (2) <https://doi.org/10.1177/20539517221110218>
- [7] Liu S., He K., Pan X., Hu Y. (2023) Impacts of intelligent transportation systems on energy conservation and emission reduction of transport systems: A comprehensive review. Green Technologies and Sustainability, 1(1), 100002, <https://doi.org/10.1016/j.grets.2022.100002>
- [8] Hussain A., Ullah K. (2024) An Intelligent Decision Support System for Spherical Fuzzy Sugeno-Weber Aggregation Operators and Real-Life Applications. Spectrum of Mechanical Engineering and Operational Research, 1(1), 177-188, <https://doi.org/10.31181/smeor11202415>
- [9] Fischer S., Szürke S. K. (2023) Detection process of energy loss in electric railway vehicles. Facta Universitatis, Series: Mechanical Engineering, Vol. 21, No. 1, doi: 10.22190/FUME221104046F
- [10] Damjanović M., Stević Ž, Stanimirović D., Tanackov I., Marinković D. (2022) Toward a smart ecosystem with automated services. Facta Universitatis, Series: Mechanical Engineering, Vol. 20, No. 1, doi: 10.22190/FUME220215012D
- [11] Hancke G. P., Silva B. D. C., Hancke Jr. GP. (2013) The Role of Advanced Sensing in Smart Cities. Sensors, 13(1), 393-425, <https://doi.org/10.3390/s130100393>
- [12] Arshi O., Mondal S. (2023) Advancements in sensors and actuators technologies for smart cities: a comprehensive review. Smart Constr. Sustain. Cities, 1(18) <https://doi.org/10.1007/s44268-023-00022-2>

- [13] Qin L., Yu, N., Zhao, D. (2018) Applying the Convolutional Neural Network Deep Learning Technology to Behavioral Recognition in Intelligent Video. *Tehnički vjesnik*, 25 (2), 528-535, <https://doi.org/10.17559/TV-20171229024444>
- [14] Tu Q., Geng G., Zhang Q. (2023) Multi-Step Subway Passenger Flow Prediction under Large Events Using Website Data. *Tehnički vjesnik*, 30 (5), 1585-1593, <https://doi.org/10.17559/TV-20230227000384>
- [15] Aydemir E., Gulsecen S. (2019) Arranging Bus Behavior by Finding the Best Prediction Model with Artificial Neural Networks. *Tehnički vjesnik*, 26 (4), 885-892, <https://doi.org/10.17559/TV-20170629201111>
- [16] Uğuz S., Büyükgökoğlu E. (2022) A Hybrid CNN-LSTM Model for Traffic Accident Frequency Forecasting During the Tourist Season. *Tehnički vjesnik*, 29 (6), 2083-2089, <https://doi.org/10.17559/TV-20220225141756>
- [17] Aydin I., Sevi M., Akin E., Güçlü E., Karaköse M., Aldarwich H. (2023) A Deep Learning-Based Hybrid Approach to Detect Fastener Defects in Real-Time. *Tehnički vjesnik*, 30 (5), 1461-1468, <https://doi.org/10.17559/TV-20221020152721>
- [18] Dinculeană D., Cheng, X. (2019) Vulnerabilities and Limitations of MQTT Protocol Used between IoT Devices, *Appl. Sci.* 9, 848, <https://doi.org/10.3390/app9050848>
- [19] Vračar Lj, Stojanović, M., Stanimirović, A., Prijic Z. (2019) Influence of Encryption Algorithms on Power Consumption in Energy Harvesting Systems, *Journal of Sensors*, Article ID 8520562, 9 pages, 2019, <https://doi.org/10.1155/2019/8520562>
- [20] Jang J.-S. R, (1993) ANFIS: Adaptive-Network-based Fuzzy Inference Systems, *IEEE Trans. On Systems, Man, and Cybernetics*, Vol. 23, 665-685
- [21] Ézsiás, L., Tompa, R, & Fischer, S. (2024) Investigation of the Possible Correlations between Specific Characteristics of Crushed Stone Aggregates. *Spectrum of Mechanical Engineering and Operational Research*, 1(1), 10-26, <https://doi.org/10.31181/smeor1120242>
- [22] Jóvér, Vivien & Gáspár, László & Fischer, Szabolcs. (2022) Investigation of Tramway Line No. 1, in Budapest, Based on Dynamic Measurements. *Acta Polytechnica Hungarica*. 19, 65-76, 10.12700/APH.19.3.2022.3.6
- [23] Szalai, Szabolcs & Bálint, Herold & Kurhan, Dmytro & Németh, Attila & Sysyn, Mykola & Fischer, Szabolcs. (2023) Optimization of 3D Printed Rapid Prototype Deep Drawing Tools for Automotive and Railway Sheet Material Testing. 10.3390/infrastructures8030043

- [24] Jover, V., & Fischer, S. (2022) Statistical Analysis of Track Geometry Parameters on Tramway Line No. 1 in Budapest. *The Baltic Journal of Road and Bridge Engineering*, 17(2), 75-106, <https://doi.org/10.7250/bjrbe.2022-17.561>
- [25] Eller, B., Majid, M. R., & Fischer, S. (2022) Laboratory tests and FE modeling of the concrete canvas, for infrastructure applications. *Acta Polytechnica Hungarica*, 19(3), 9-20, <https://doi.org/10.12700/APH.19.3.2022.3.2>
- [26] Németh, A., & Fischer, S. (2018) Investigation of glued insulated rail joints with special fiber-glass reinforced synthetic fishplates using in continuously welded tracks. *Pollack Periodica Pollack Periodica*, 13(2), 77-86, <https://doi.org/10.1556/606.2018.13.2.8>
- [27] European Committee for Standardization (CEN) (2018) EN 14363:2016+A1:2018. Railway applications – Testing and simulation for the acceptance of running characteristics of railway vehicles – Running behaviour and stationary tests. Brussels: CEN