Naturalistic Driving Studies Data Analysis Based on a Convolutional Neural Network

Jamal Raiyn[1] and Galia Weidl[2]

[1]Computer Science Department, Al Qasemi Academic College, Baqa Al Ghariah, Israel

[2]Connected Urban Mobility, Faculty of Engineering, Technical University of Applied Sciences, Aschaffenburg, Germany

Keywords: Convolutional Neural Network, Deep Learning, Autonomous Vehicle, Anomalous Data.

Abstract: The new generation of autonomous vehicles (AVs) are being designed to act autonomously and collect travel data based on various smart devices and sensors. The goal is to enable AVs to operate under their own power. Naturalistic driving studies (NDSs) collect data continuously from real traffic activities, in order not to miss any safety-critical event. In NDSs of AVs, however, the data they collect is influenced by various sources that degrade their forecasting accuracy. A convolutional neural network (CNN) is proposed to process a large amount of traffic data in different formats. A CNN can detect anomalies in traffic data that negatively affect traffic efficiency and identify the source of data anomalies, which can help reduce traffic congestion and vehicular queuing.

1 INTRODUCTION

The rapid growth of smart cities, with their severe traffic congestions, road safety issues and environmental pollution problems, presents autonomous vehicles with challenges, including an increased risk of human operating error. Modern autonomous vehicles (AVs) are equipped with multiple sensors, such as cameras, radars and LIDAR, which help them to better perceive their surroundings and plan travel routes. These sensors generate a massive amount of data and one challenge to autonomous vehicles is how to manage the massive amount of data that is collected by these various devices (Arena and Pau, 2019). AVs use the data to operate independently, communicating, negotiating and making decisions. These attributes make the new generation of AVs one of the key areas of applications for artificial intelligence (Li at et 2018). AVs face many challenges, such as anomalous data that is caused by geographical factors and cyber-attackers. Some road data arrive in an incomplete or falsified form (Xiang et al, 2019; Qin et al, 2021) and can result in abnormal conditions on urban roads (Raiyn, 2021; Zhang eta l, 2020). Currently, many research enterprises and universities around the world have taken the initiative to design AVs with artificial Intelligence, so that they can perform better than the human brain. A convolutional neural network (a CNN or ConvNet) (Yamashita et al, 2018) is designed to mimic the working principle of the human brain and is trained with large Big Data sets to perform various tasks. To manage road networks, such a network is proposed here, with the goal of detecting road sections that are overloaded because they have been given inappropriate information. A CNN is a specialized form of artificial intelligence technology that analyzes input data which contain some form of spatial structure. CNNs are used to solve visual computational problems, such as those associated with self-driving cars, robotics, drones, security, medical diagnoses, and agriculture. In this paper, A CNN (Arena and Pau, 2019) is proposed. CNNs are considered to be one of the most important forms of artificial intelligence. They consist of a computational elements (neurons) heavily connected to each other, which process perceptual data from the surrounding environment, such as images of road traffic from satellites and drones, and information from sources based on vehicle-to-vehicle (V2V) and vehicle-to-cloud (V2C) communication and from digital resources like Google Maps.

1.1 Literature Review

The field of anomaly detection has been widely researched (Santhosh et al, 2020; Aradi, 2022).

Raiyn, J. and Weidl, G.
Naturalistic Driving Studies Data Analysis Based on a Convolutional Neural Network.
DOI: 10.5220/0011839600003479
In *Proceedings of the 9th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2023)*, pages 248-256
ISBN: 978-989-758-652-1; ISSN: 2184-495X

Anomaly detection approaches are typically divided into two types: model-based and data analysis-based. Model-based approaches mostly use algorithms that are very accurate, such as machine learning schemes (Xiang et al, 2019), whereas the approaches based on data analysis usually use statistical measurements. On urban roads, anomalies cause discomfort to drivers and have a negative impact on traffic efficiency (Silva et al, 2018). When traffic accidents occur and there is a congestion, the resulting traffic flow becomes abnormal. In an AV network, anomalies are caused by traffic accidents, bad weather, road work, and repeated lane changing attempts. In addition, there are other challenges (Mishra et al. 2021) faced by AVs, such as noise and interference, which are further sources of anomalies in traffic flow. AVs collect various types of data via onboard devices and communication with devices on the Internet of Things (Aradi, 2022). Both the onboard devices and the AV communications protocols are affected by interferences and delays (Niknam et al, 2018). AV positioning data is collected by the Global Navigation Satellite System (GNSS) (Raiyn, 2017), whose satellites are mainly located in medium earth orbits (MEO). The signals transmitted by a satellite propagates through the atmosphere, where they are subject to delays caused by ionospheric and tropospheric media. At ground level, multipath effects, namely the reception of signals that are reflected from obstacles such as buildings surrounding the receiver, can occur, causing one of the largest types of errors, one that is also difficult to model, as it depends strongly on the receiver environment. Increased delays may affect the performance metrics of a positioning terminal, which are characterized in terms of availability, accuracy, and integrity. Delays may also be caused by the weak performance of network equipment and differences in equipment attributes. Many untargeted transmitted signals can even interfere with transmitted signals (Raiyn, 2021). Now, there are new challenges that have yet to be considered, such as communication between AVs in heterogeneous wireless networks. Heterogeneity can cause delays in communication between AVs. For AV communication in differentiated wireless networks that provide differing quality of service (QoS), middleware is needed to make adaptations. Another challenge faced by AVs is identifying road sections with a high degree of noise. Furthermore, the devices in an AV that has been involved in an accident may stop working and render it unable to communicate with surrounding vehicles. These and many other factors, cause anomalies in data that can increase daily traffic congestion. To detect road traffic anomalies, various forecasting schemes have been proposed (Yan et al, 2018). Recently, some deep learning approaches have been introduced to predict urban traffic flow. The most established algorithm among various deep learning models is using CNNs, a class of artificial neural networks that has been the dominant technology for computer vision tasks since it first started producing astonishing results. Deep neural networks were proposed to predict traffic condition on highways by considering spatio-temporal correlations in traffic data attributes. To make more accurate forecasts, an advanced CNN can also incorporate data sources, such as weather forecasts, and police reports. In (Yamashita et al, 2081), a CNN was designed to automatically and adaptively learn a spatial hierarchy of features through backpropagation by using multiple building blocks, such as convolutional layers, pooling layers, and fully connected layers. A CNN can exploit the non-linear regularities of network traffic, providing significant improvements with respect to the mean absolute and standard deviation of data (Mozo et al, 2018). The performance of several models was compared using different accuracy measurement methods (e.g., Root Mean Square Error [RMSE] and Mean Absolute Percentage Error [MAPE]). The results indicate the good performance of the Prophet and CNN models (Yan et al, 2018).

1.2 Problem Description

Various classical forecast schemes have been proposed to manage road traffic flow. These schemes perform well in historical data management, however, they are lacking in real-time forecasting. Traffic flow is becoming more complex, and information about it has evolved from a single format to a conglomeration of formats providing very large datasets, known as Big Data. To process Big Data, a new scheme is proposed which is based on computational intelligence. Furthermore, the forecasting schemes are designed to process dynamic traffic flow instead of fixed traffic flow. Classical computing schemes, performed short-term forecasting for a given time slot and considered traffic flow as a fixed entity, represented by 0 for busy and 1 for free. However, real-time traffic flow dynamic can change every millisecond. The proposed CNN considers dynamic traffic flow, represented by 00 for busy, 11 for free and 01 for busy changed to free and 10 for free changed to busy.

1.3 Research Contribution

The main contributions of the model described in this article are as follows: Firstly, it can detect anomalies in traffic data. Second, it can extract the features of traffic data and identify the sources of anomalous data (Raiyn, 2022). Third, it can perform a short-term travel forecasts by applying CNNs, and fourth, it makes it possible to compare the accuracy of results through measurement methods such as RMSE and MAE. The results are analyzed and the best method for prediction is selected. This paper is organized as follows: Section 2 gives an overview of deep learning. Section 3 describes the methodology and discusses the development of the CNN. Section 4 concludes the discussion and point out directions for future research.

2 DEEP LEARNING TECHNOLOGY

Deep-learning (DL) methods are some of the most useful tools in the area of machine learning (Li et al, 2018; Xiang et al, 2019) (Berman et al, 2019). A DL system learns the features of a dataset and then combines them to achieve a specific goal. In this paper, a DL method is proposed for the early detection of traffic anomalies (Nguyen et al, 2018). It is composed of two main phases: training and testing. The system proposed here is composed of four phases: (1) preparation of the dataset, (2) a training phase, (3) a testing phase, and (4) a performance metrics phase (Wang et al, 2019; Raiyn, 2017). The deep learning (DL) system, in general, is structured in three main parts: an input layer, a hidden layer, and an output layer. The input layer for DL consists of a large amount of data that is received from different sources. The big dataset for traffic modeling is diverse and comes from variety of devices and systems, such as cameras, LIDAR, sensors, and GNSSs (Raiyn, 2017). The neurons have an important influence on the learning ability of the algorithm; too few can lead to insufficient learning, and too many can lead to overfitting. The output layer is responsible for exporting the values, or the vectors of the values, that correspond to the format required for the problem, and it presents the results visually based on measurements of statistical error.

2.1 Convolutional Neural Network

CNNs are a type of deep learning model for processing data, one- dimensional, two- dimensional and three- dimensional (Ensafi et al, 2022; Cordeiro et al, 2021). They are designed to automatically and adaptively learn the spatial hierarchies of features, from low- to high-level patterns. They are typically composed of three types of layers: convolutional, pooling, and fully connected layers. The first two, convolutional and pooling layers, perform feature extraction, whereas the third, the fully connected layer, maps the extracted features into final output, such as classification. The architecture of a CNN consists of three distinct types of layers: a convolutional layer, pooling layer, and a classification layer. The convolutional layers are the core of the CNN. The weights define a convolutional kernel applied to the original input, a small window at a time, called a receptive field. The results from the application of these filters across the entirety of the input are then passed through a non-linear activation function, typically a ReLU, to produce a feature map. The three layers in the architecture of a CNN are as illustrated in Figure 1.

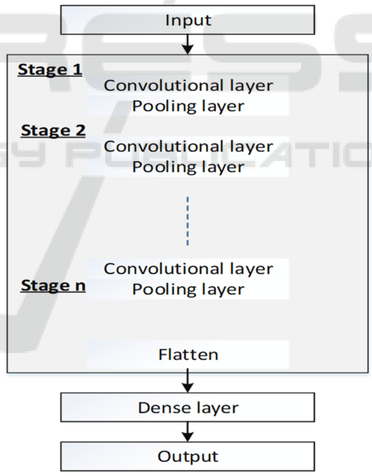

Figure 1: CNN architecture.

3 METHODOLOGY

The aim of this study is to make time-series forecasts with the help of different models, such as, exponential smoothing and CNNs. In this investigation, the model with the most accurate results qualifies as the best model (Alzubaidi et al, 2021). The first step in anomaly detection is to define normal traffic on a section of urban road and then to flag as anomalies any observations that do not fit this normal pattern. Finding these patterns is the main challenge in detecting anomalies in urban road traffic. This section describes model based on a CNN that can be applied to all types of normally trafficked roads. Through the analysis, the CNN extracts anomalous data. An important step is cleaning and formatting the data. Travel data are collected by different devices and are influenced by human and environmental factors. *Data cleaning* is the process of modifying the data to ensure that the datasets are free of irrelevancies and incorrect or incomplete travel observations. The best computational data formats have several useful features; for instance, they are easy for computers to parse, easy for people to read, and widely usable by other tools and systems. The travel data input for this study came from various smart devices in all kinds of formats. This dataset was cleaned to improve the efficiency of the data analysis and the quality of the results. Due to noise and environmental factors that influence data communication, some of the datasets collected were not complete; in this case, data cleaning involved identifying the fields from which data were missing and then properly compensating for them. An interactive exploratory travel dataset was useful for representing errors in rea-time. The deep learning scheme extracted the attributes from the input data. The attributes were classified and assigned scores.

$$y_{AD} = \sum x_n w_n + error \qquad (1)$$

Here, x_n is the input signal, w_n is the weight corresponding to each input signal, and y_{AD} is the output signal. F_{DL} is the activation function, the means of calculating the sum of the data coming from the input.

3.1 Convolutional Neural Network 2D

AV systems use CNNs to detect the behavior of urban road traffic. CNN can handle the urban road network in old cities. In this case, the input data consisted of the geographical locations of all the data sources in an urban area that was powered by online map services, such as Google Maps, as illustrated Figure 2a. The images in the figure are raw snapshots from a traffic congestion map for an urban area in Kaboul town, whose roads are constructed according to Palestinian building standards, which means they are narrow and crooked. The raw snapshots for this area mainly cover urban arterial roads; each is 400 pixels wide and 400 pixels high. The snapshots were retrieved during morning rush hours between 8:00 and 10:00 in June 2, 2022, through a free API provided by an online map service provider. These snapshots were the initial source of traffic congestion data. The procedure included cleaning the data as illustrated in Figure 2b, c and d.

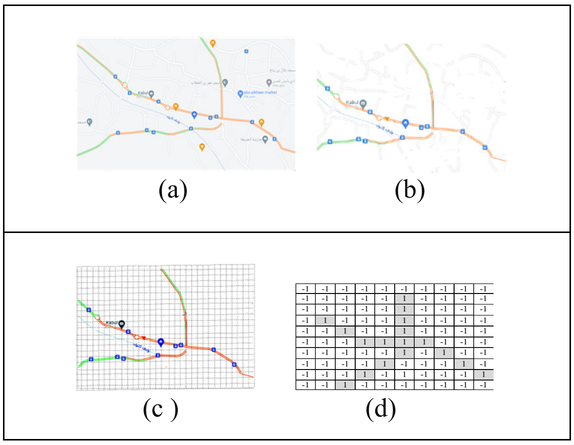

Figure 2: Preparing the input data.

The second phase included a hidden phase and an output phase as explained below.

3.2 Forecasting Model

In this section, the framework for representing the traffic congestion data for a road network is described. The proposed representation framework consists of two steps. The first step, which segments the original traffic congestion is prepared according to the data science life cycle. The second step reduces all the values in each grid using CNN one- and two-dimensional operations to reduce in a manner similar to image down-sampling. To detect data anomalies, two approaches are proposed: use of the exponential moving average and use of a CNN.

3.2.1 Classical Time-Series Forecasting

In (Li, et al, 2018), the exponential smoothing scheme is used for time- series forecasting. The aim of this scheme is to detect data anomalies in road flow traffic. The exponential smoothing scheme lends greater weight to the most recent travel observations, so that the older the observation, the less it affects the forecast.

$$\tilde{y}_{t+1} = \theta_0 y_t + \theta_1 y_{t-1} + \theta_2 y_{t-2} + \ldots \qquad (2)$$

Where $\theta_i = \alpha(1-\alpha)^i$, α is the smoothing constant, and $0 < \alpha \le 1$.

Hence the forecast scheme results in

$$\tilde{y}_{t+1} = \alpha y_t + \alpha(1-\alpha) y_{t-1} + \alpha(1-\alpha)^2 y_{t-2} + \ldots \qquad (3)$$

which can be easily described as follows:

$$\tilde{y}_{t+1} = \alpha y_t + (1-\alpha)\tilde{y}_t \quad (4)$$

Exponential smoothing allows us to update the prediction when a real time observation is available. Anomaly detection based exponential smoothing is performed based on a travel data analysis. Exponential smoothing uses statistical error measurements to characterize the integrity and consistency of the travel data. Data anomalies associated with road accidents are characterized by alterations in travel speed which starts to decrease in the upstream direction and increase in the downstream direction. Furthermore, significant differences become evident between travel speeds and standard deviations.

3.2.2 Convolutional Neural Network

- Input layer

The input for the model is the time- series data for road traffic. The lengths of input and output time intervals can be expressed as F and P, respectively. The model input can be written as:

$$x^i = [m_i, m_{i+1}, \ldots, m_{i+P-1}], i \in [1, N-P-F+1] \quad (5)$$

Where I is the sample index, N is the length of the time intervals, and m_i is a column vector representing the traffic speed of all the road sections in a transportation network within one time unit. In some highways in Middle East countries, the average speed is 90 km. Travel observations that exceed that threshold are represented by a value between 0 and -1; otherwise, they are represented by a positive value between 0 and 1 as illustrated in Figure 3. The CNN is applied to detect anomalies in the road traffic data. In this case, only the negative values are considered and at the same time present the travel data that are smaller than the threshold.

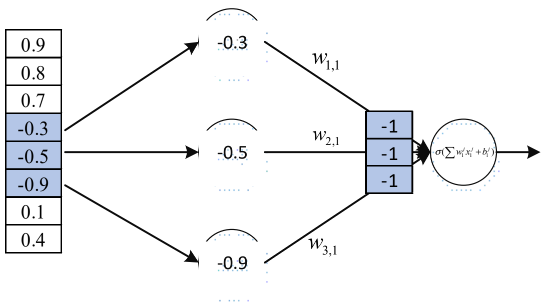

Figure 3: Features extraction.

The extraction of traffic features involves a combination of the convolutional and pooling layers. The output of the first convolution and pooling layers can be written as:

$$o_1^j = pool(\sigma(W_1^j x_1^j + b_1^j)), j \in [1, c_1] \quad (6)$$

and the output of the last convolutional and pooling layers can be written as

$$o_n^j = pool(\sigma(W_n^j x_n^j + b_n^j)), j \in [1, c_n]$$

where σ is the activation function. In the prediction, the features learned and outputted by traffic feature extraction are concatenated into a dense vector that contains the final and the highest-level features of the transportation network input. The dense vector can be written as

$$o_L^{flatten} = flatten([o_L^1, o_L^2, \ldots, o_L^j]), j = c_L \quad (7)$$

where L is the depth of CNN. Finally, the vector is transformed into output through a fully connected layer. The output can be written as:

$$\tilde{y} = W_f o_L^{flatten} + b \quad (8)$$

$$= W_f(flaten(pool(\sigma(\sum_{k=1}^{c_L-1}(W_L^j x_L^k + b_L^j))))) + b_f \quad (9)$$

where W_f and b_f are the parameters of the fully connected layer, and $\tilde{y}$ represents the predicted network-wide data anomalies. The CNN uses convolutional filters on its input layer and obtains local connections only where local input neurons are connected to an output neuron (in the convolutional layer). Hundreds of filters are sometimes applied to the input, and the results are merged in each layer. One filter can extract one traffic feature from the input layer and, therefore, hundreds of filters can extract hundreds of traffic features as illustrated in Figure 4. The fully connected layer expresses the negative values, that represent the anomalies in each road section.

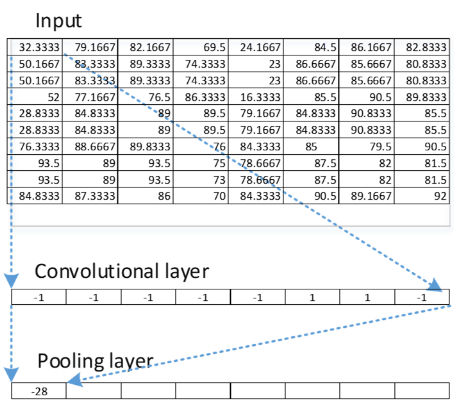

Figure 4: Fully connected layer.

3.3 Comparative Methods

To manage road traffic flow, two approaches are proposed.

- The first scheme is the exponential smoothing scheme. Used for short-term forecasting, this scheme is based on both historical and real-time travel data. It calculates a short-term forecast for a given period of time (e.g. 15 minutes, a day, a week) without considering changes in road traffic flow after that. The output of the exponential smoothing scheme is a value that represents the speed on the road. The result is saved in 0, 1 format, which described the road traffic as busy or free respectively. The second approach involves CNNs with one and two dimensions. A 1D CNN is used to handle time- series road traffic; the 2D CNN can handle images such as maps.
- An exponential smoothing scheme needs a large amount of input data to make forecasts, however, the CNN can use an image of a road section to perform traffic analysis and make forecast.
- The input travel data should be structured for an exponential smoothing scheme; however, the advantage of a CNN (1D, 2D, and 3D) is that it can process any data format.

The CNN can identify sources of abnormalities. Detecting abnormalities is a complex task due to the variability of activity from one case to another, and due to the absence of standardized datasets representing normal and abnormal activity. For anomaly detection, the CNN is trained on data representing the activity and behavior of the sources of abnormalities. The initial set of data, collected from time- series representing the usual behavior of road traffic, serves as a baseline for training the novelty detection models. Road observations recognized as anomalies are explored further to determine the sources of the abnormalities, as depicted, and the CNN is trained to identify further abnormalities based on the behavior and activity of other sources.

4 EVALUATION OF FORECASTING SCHEMES

In general, AVs collect a variety of traffic data. Low-quality data can lead to traffic congestion and collisions. Furthermore, data reception may be incomplete due to the urban noise produced by network tunnels. The raw traffic data used in this project was mainly obtained from smart phones in 2.5-minute cycles. When a statistical analysis of the original data was carried out, it was found that it involves two main defects; the dataset was incomplete, and the data contained noise. Urban roads are divided into sections each 300 meters long. The data were processes by several operations: statistical measurements were used to detect incomplete data, and unnecessary material such as noise was removed (Alrajhi et al, 2019). The loss function was a customized mean squared error (MSE), as defined in Eq. 10. *lMSE* which is described in Eq. 11, was applied to calculate the mean squared error at different congestion levels with different priorities. *pMSE* in Eq. 12 was applied to calculate the mean squared error at different congestion levels with different positioning.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2 \tag{10}$$

$$lMSE = \frac{1}{n}\sum_{i=1}^{n} l_i (y_i - \tilde{y}_i)^2 \tag{11}$$

$$pMSE = \frac{1}{n}\sum_{i=1}^{n} p_i (y_i - \tilde{y}_i)^2 \tag{12}$$

The cost function is a special type of functions that helps to minimize error and to approach as close as possible the expected output. It uses two parameters to calculate error: one is an estimated output of the CNN model (also called the prediction) and other is the actual output. The mean squared error is considered one of the most familiar loss functions as it is much like the least square loss function. It directly calculates the difference between the predicted result and the true label, which is denoted as in Eq.13. A basic working out of error can be formed with the input data when there is an actual value and a predicted value. Error can be characterized as the difference between the predicted value and the actual value. The loss function in the first convolutional and pooling layers can be formulated as

$$\mathrm{J}(\theta_0, \theta_1) = \frac{1}{2n}\sum_{i=1}^{n}(h_\theta(x^{(i)}) - y^{(i)})^2 \tag{13}$$

The loss function of all the convolutional and pooling layers can be formulated as

$$\mathrm{J}(\theta_{\text{n-1}}, \theta_n) = \frac{1}{2n}\sum_{conv}\sum_{i=1}^{n}(h_\theta(x^{(i)}) - y^{(i)})^2 \tag{14}$$

The loss function of all the convolutional and pooling layers for all the road section can be formulated as

$$J(\theta_{n\text{-}1},\theta_n)=\frac{1}{2n}\sum_{section}\sum_{conv}\sum_{i=1}^{n}(h_\theta(x^{(i)})-y^{(i)})^2 \qquad (15)$$

where $h_\theta(x)$ is the prediction that is closest to the actual value, y. To reduce forecasting error, the CNN uses a self-management strategy to control the output of each convolutional layer. The calculation of loss function is related to road sections. In training the CNN, weights are selected that capture a desired input-output relationship. This training objective can be framed as a minimization of a loss function, which quantifies the difference between the output of the network and the ground truth values from the training set. The forecast of the CNN are given in terms of traffic speed on different road sections, and the mean squared errors (MSEs) are employed to measure the distances between the forecasts and actual traffic speed. Minimizing MSEs is taken as the training goal of the CNN. MSE can be written as

$$MSE=\frac{1}{n}\sum_{i=1}^{n}(y_i-\tilde{y}_i)^2$$

when the model parameters are set , the optimal values of θ can be determined according to the *standard back propagation algorithm,* which is used in other studies of CNN:

$$\begin{gathered}\min J_1(\theta_0,\theta_1)\\ \min J_2(\theta_1,\theta_2)\\ \vdots\\ \vdots\\ \vdots\\ \min J_n(\theta_{n-1},\theta_n)\end{gathered} \qquad (16)$$

where,

$$\theta=\underset{\theta}{\operatorname{argmin}}\frac{1}{\mathrm{n}}\sum_{i=1}^{N}(\tilde{y}_i-y_i)^2 ,$$

$$\theta=\underset{\theta}{\operatorname{argmin}}\frac{1}{\mathrm{n}}\| W_f\sigma_L^{flatten}+b_f-y\|^2 ,$$

$$\theta=\underset{\theta}{\operatorname{argmin}}\frac{1}{\mathrm{n}}\| W_f(flatten(pool(\sigma(\sum_{k=1}^{C_{l-1}}(W_L^j X_L^k+b_L^j)))))+b_f-y\|^2$$

The quality of a forecast depends on the training of the neural network $F(X;\theta)$, with the aim of finding a suitable set of parameters θ so that the model can achieve good performance. The task of training a neural network is equivalent to optimizing the loss function by back-propagation iteration. More precisely, a loss function outputs a scalar value which is regarded as a criterion for measuring the difference between the predicted result and the true label for one sample. During training, our goal was to minimize the scalar value for m training samples (i.e., the cost function).

- Detecting of the source of data anomalies

The CNN was capable of extracting a variety of data from a given 2D image, such as road traffic data, AV positioning, and anomalous data and their sources. Anomalies in the data may stem from geographical factors or experimental or human error. To elicit good performance from the CNN, the training dataset should be free from anomalies. An exponential smoothing scheme employing statistical functions was used to detect anomalous data. Simple statistical functions were applied to detect univariate anomalous feature values in the data sets as illustrated in Figures 5. An anomaly score was computed for all observations; if the score was greater than a given threshold value, the data were considered anomalous. Figure 6 show data with missing values that were collected with a uBlox device. Figure 7 show the detected anomalies in road traffic.

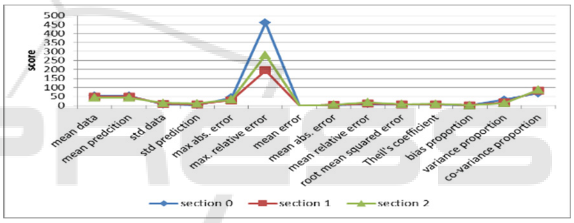

Figure 5: Statistical measurment error.

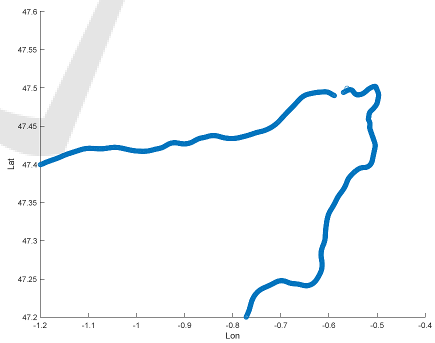

Figure 6: Missing data.

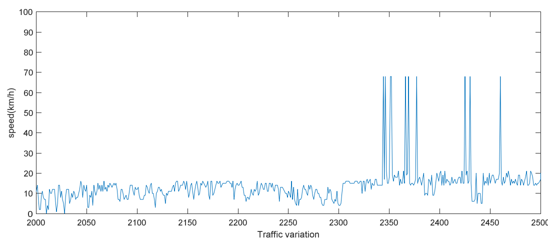

Figure 7: Variation in travel time.

5 CONCLUSION AND FUTURE WORK

This paper proposes a CNN scheme that can predict road traffic speed based on extracted features from 2D images. The main goal of this project was to detect anomalies in road data and their sources. The discussion began by introducing the new challenges that faces modern AVs. This was followed by an overview of applications of artificial intelligence in AVs, such as deep learning algorithms and more specifically CNNs, including 1D, 2D, and 3D convolution processing alternatives. The development of a new generation of AVs equipped with various sensors and Internet of Things devices calls for new data management schemes. The main drawback of traditional traffic forecasting schemes is that they cannot manage data in different formats. Furthermore, traditional forecasting schemes process a traffic road network in its entirety, which increases the processing time. On the other hand, the application of CNNs in road traffic detection has demonstrated significant improvements over traditional approaches. A CNN can process three forms of input data such 1D, 2D and 3D. The advantages of using a CNN are that it can process data in many independent layers and each layer can be optimized.

ACKNOWLEDGMENTS

This study has been supported by the Project 101076165 — i4Driving within Horizon Europe under the call HORIZON-CL5-2022-D6-18 01-03, which is programmed by the European Partnership on ‘Connected, Cooperative and Automated Mobility’ (CCAM).

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

DATA AVAILABILITY

The majority of the datasets used in this paper are publicly available. Private datasets can be furnished upon request.

REFERENCES

Arena, F., Pau, G.: An Overview of Vehicular Communications, future internet,11,27. pp.112. (2019).

Li, J., Cheng, H., Guo H., Qiu, S.: Survey on Artificial Intelligence for Vehicles”,
Automotive Innovation, Springer, (2018). https://doi.org/10.1007/s42154-018-0009-9

Xiang, W., Huang, T., Wan, W.: Machine learning based optimization for vehicle-to-infrastructure Communications, Future Generation Computer Systems 94, 488–495. (2019).

Qin, H., Yan, M., Ji, H.: Application of Controller Area Network (CAN) bus anomaly detection based on time series prediction, Vehicular Communications, 27. (2021).

Raiyn, J.: Classification of Road Traffic Anomaly Based on Travel Data Analysis, International Review of Civil Engineering (IRECE), 12, 4. (2021).

Zhang, S., Li, S., Li, X., Yao, Y.: Representation of Traffic Congestion Data for Urban Road Traffic Networks Based on pooling Operations, Algorithms, 13(84), 1-18. (2020).

Yamashita, R., Nishio, M., Do, R.K.G., Togashi, K.: Convolutional neural networks: an overview and application in radiology, Insights into Imaging,9, pp. 611-629. (2018).

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaria, J., Fadhel, M.A., Al-Amidie, M., Farhan, L.: Review of deep learning: concepts, CNN architectures, challenges, applications, future Directions, Journal of Big Data, 8(23), 1-74. (2021).

Silva, N., Shah, V., Soares, J. Rodrigues, H.: Road Anomalies Detection System Evaluation, Sensors, 18, 1-20. (2018).

Santhosh, K.K., Dogra, D.P., Roy, P.P.: Anomaly Detection in Road Traffic Using Visual Surveillance: A Survey, ACM Computing Surveys, 6(53), 1-26. (2020).

Mishra, A., Cohen, A., Reichherzer, T., Wilde, T.: Detection of data anomalies at the edge of pervasive IoT Systems, Computing, Springer, (2021).

Aradi, S.: Survey of Deep Reinforcement Learning for Motion Planning of Autonomous Vehicles, IEEE Transactions On Intelligent Transportation Systems, Vol. 23, No. 2, 740-759. (2022).

Berman, D.S., Buczak, A.L., Chavis, G.S. Corbett, C.L. A Survey of Deep Learning Methods for Cyber Security, Information, 10, 122. 1-35. (2019). doi:10.3390/info10040122

Nguyen, H., Kieu, L.-M., Wen, T., Cai, C.: Deep learning methods in transportation domain: a review, IET Intelligent Transport System, 12, 9.998-1004. (2018).

Alrajhi M., Kamel, M.: A Deep-Learning Model for Predicting and Visualizing the Risk of Road Traffic Accidents in Saudi Arabia: A Tutorial Approach, International Journal of Advanced Computer Science and Applications, Vol. 10, No. 11, 475-483. (2019).

Wang, J., Cehn, R., He, Z.: Traffic speed prediction for urban transportation network: A path based deep learning approach, Transportation Research Part C, 100. 372-385. (2019).

Raiyn, J.: Developing Vehicle Locations Strategy on Urban Road, Transport and Telecommunication, 18. (4). 253–262. (2017).

NIknam, S., Natarajan, B., Barazidh, R.: Interference Analysis for Finit-Area 5G mm Networks Considering Blockage Effects, IEEE Access, Vol. 6. 23470-23479. (2018).

Mozo, A., Ordozgoiti, B., GoÂmez-Canaval, S.: Forecasting short-term data center network traffic load with convolutional neural networks, PLoS ONE 13(2): 1-31. (2018). e0191939. https://doi.org/10.1371/journal.pone.0191939.

Ensafi, Y., Amin, S.H., Zhang, G., Shah, B.: Time-series forecasting of seasonal items sales using machine learning –A comparative analysis, International Journal of Information Management Data Insights,2, 1-16. (2022).

Cordeiro, J.R., Raimundo, A., Postolache, O., Sebastião, P.: Neural Architecture Search for 1D CNNs—Different Approaches Tests and Measurements, Sensors, 21. 1-23. (2021). https://doi.org/10.3390/s21237990.

Raiyn, J. Using Dynamic Market- Based Control for Real-Time Intelligent Speed Adaptation Road Networks, *Advances in Science, Technology and Engineering Systems Journal,* Vol. 7. No. 4. 24-27.