Article

Feasibility of Neural Network Based Virtual Sensor for Vehicle Unsprung Mass Relative Velocity Estimation

Eldar Šabanovič[1],*, Paulius Kojis[1], Šarūnas Šukevičius[2], Barys Shyrokau[3], Valentin Ivanov[4], Miguel Dhaens[5], and Viktor Skrickij[1]

[1] Transport and Logistics Competence Centre, Transport Engineering Faculty, Vilnius Gediminas Technical University, Saulėtekio al. 11, 10223 Vilnius, Lithuania; eldar.sabanovic@vilniustech.lt (E.S.); paulius.kojis@vilniustech.lt (P.K.); viktor.skrickij@vilniustech.lt (V.S.)

[2] Department of Mobile Machinery and Railway Transport, Transport Engineering Faculty, Vilnius Gediminas Technical University, Plytinės str. 27, 08101 Vilnius, Lithuania; sarunas.sukevicius@vilniustech.lt (Š.Š.)

3 Department of Cognitive Robotics, Delft University of Technology, 2628CD Delft, The Netherlands; b.shyrokau@tudelft.nl (B.S.)

4 Automotive Engineering Group, TU Ilmenau, 98693 Ilmenau, Germany; valentin.ivanov@tu-ilmenau.de (V.I.)

5 Tenneco Automotive Europe, 3800 Sint-Truiden, Belgium; MDhaens@Tenneco.com (M.D.)

*   Correspondence: eldar.sabanovic@vilniustech.lt.

Abstract: With the automotive industry moving towards automated driving, sensing is becoming an increasingly important part of enabling technology. The virtual sensors allow data fusion from various vehicle sensors and provide a prediction for measurement that is hard or too expensive to measure in another way or in the case of demand on continuous detection. In this paper, virtual sensing is discussed for the case of vehicle suspension control, where information about the relative velocity of the unsprung mass for each vehicle corner is required. The corresponding goal can be identified as a regression task with multi-input sequence input. The hypothesis is that the state-of-art method of Bidirectional Long-Short Term Memory (BiLSMT) can solve it. In this paper, a virtual sensor has been proposed and developed by training a neural network model. The simulations have been performed using an experimentally validated full vehicle model in IPG Carmaker. Simulations provided the reference data which was used for Neural Network (NN) training. The extensive dataset covering 26 scenarios has been used to obtain training, validation and testing data. The Bayesian Search was used to select the best neural network structure using root mean square error as a metric. The best network is made of 167 BiLSTM, 256 fully connected hidden units and 4 output units. Error histograms and spectral analysis of the predicted signal compared to the reference signal are presented. The results demonstrate the good applicability of neural network-based virtual sensors for the estimation of vehicle unsprung mass relative velocity.

Keywords: virtual sensor; automotive control; active suspension; vehicle state estimation; neural networks; deep learning; long-short term memory; sequence regression

1. Introduction

Nowadays, the automotive industry focuses a lot on Automated Driving (AD) as a very promising solution to improve safety and comfort. The main functional components related to AD include perception, decision-making, and vehicle control. New generation vehicles will need more information to accomplish these three tasks and ensure safe driving without human involvement. The required data can be gained using standard and novel sensors, Vehicle to Everything (V2X) communication, and sensor fusion.

Sensors are crucial components needed for AD, as they provide the data required to perceive the environment and vehicle state estimation [1]. The first group include laser imaging detection and ranging (LIDAR), radio detection and ranging (Radar) [2], ultra-

sonic distance sensor and camera. These sensors are placed outside the vehicle, and measurement accuracy may be affected if covered by dirt, snow or ice. Cameras can be installed inside the cabin, which may prevent from disadvantages mentioned above. They can be applied for traffic objects detection and tracking, for vision-based localisation and navigation, to capture textures and colours. Usage of stereo and infrared cameras can expand their uses for geometric parameter estimation, capturing objects in dark lighting [3–5].

The second group of sensors used for vehicle state estimation involve a global navigation satellite system (GNSS), inertial measurement unit (IMU), accelerometers, displacement sensors, and wheel encoders. These sensors gather data about geolocation, vehicle position, angular rates and body's accelerations. In addition, some novel sensors for wheel load reconstruction have been recently proposed [6, 7].

To pre-process and filter the data, sensor fusion of multiple sensors data provides more reliable and accurate input data. For example, combining GNSS and IMU provides vehicle information about global position and velocity. Camera and LIDAR or Radar fusion provides 3D environment representation [1,2].

In recent years, the field of virtual sensing has been intensively investigated. The Virtual Sensors (VS) use existing signals from other on-board physical sensors combined with mathematical models or computing structures to approximate the system's state and create virtual signals of derived physical quantities that cannot be measured directly [8], e.g. indexes of performance and efficiency. Also, VS can be used in the areas where the installation of a physical sensor is challenging. For example, pressure measurement in shock absorber or force measurement in wheel carrier/bearing [9].

VS technology requires additional development costs but reduces repetitive maintenance costs [8]. At the same time, a decrease in the system's physical parts increases the overall reliability. Diagnostic applications could be incorporated by observing and predicting the system's state in advance or detecting machine degradation [10–12, 13]. In the case of synthetic data use for VS, preparing a mathematical model requires high competencies and skills. A mismatch between the model and the actual system leads to failure and high inaccuracy. When experimental data is used for VS training, computing structures on which sensors operate require accurate measurements and large data sets that cover as many real-life cases as possible, including the rare ones. The insignificant error can generate a significant drift in the estimated signals [8,14].

VS are classified according to their development approach [8,10]: *i)* measurement characteristics-based, *ii)* modelling methods-based and *iii)* application purpose-based. Measurement characteristics-based VSs are used to represent the system's steady-state or transient measurements. Steady-state modelling is defined by the instant response to input variables and moderate changes in measured values compared to the system's dynamics. Transient state type reacts slower due to complexity but allows a faster rate of change in input values. Modelling methods-based VS can be divided into data-driven, model-based or rule-based, considering modelling methods. Data-driven VSs are derived from historical data gathered by physical sensors. Model-based sensors operate using fundamental physical laws and mathematical relations between variables. Lastly, the rule-based VS utilises both approaches and relies on physical parameters and empirical models. Following application purposes, VSs are either used as backup/redundancy or observing elements. There is the possibility to validate that physical sensors detect faults and calibrate them.

Virtual sensing is intensively used for automotive applications. For example, for passenger thermal comfort, tire pressure monitoring system, powertrain applications, sprung mass state estimation [15, 16] and others. With AD development, the role of VS will only increase as power-efficient embedded computing devices will be available. VS may be a key technology of advanced control algorithm enabling customisation of vehicle-human interaction.

The vehicle-human interaction by AD needs to be carefully designed and evaluated, taking into account different aspects. As part of the task, this requires revising ride comfort and safety of the vehicles in urban and countryside conditions. The suspension design

has a crucial influence on the ride comfort and handling of the vehicle [17]. The tuning of the passive suspension has limited applicability to improve the vehicle's ride quality, as comfort and handling are conflicting objectives [18,19]. As a solution, semi-active or active suspension controlled by specialised control algorithms should be used.

During the last decades, many strategies for suspension control have been developed. The majority of them use the velocity of sprung mass and velocity of unsprung masses as input parameters. In commercial systems, velocities are evaluated using integrated data from IMU sensor placed on vehicle sprung mass and integrated/differentiated data from accelerometer and displacement sensor placed on vehicle unsprung mass. The combination of accelerometer and displacement sensor is often used for unsprung mass velocity estimation, as there is much noise in the measurement data, and it is not easy to obtain acceptable results using only one sensor. The development of virtual sensors may solve the need for physical sensors on unsprung mass. Only a few works in this field appeared in recent years.

One of the first works was Milanese et al. [20]. The authors investigated the problem of designing suitable linear virtual sensors with a focus on the estimation of the relative vertical position and velocity between chassis and wheel, using the data from the accelerometer placed on the chassis or wheel. For the task, the Direct Virtual Sensor design technique has been proposed. In 2014, Pletschen and Badur [21] presented a new nonlinear suspension state estimation approach based on Kalman Filter theory and Takagi-Sugeno modelling. Wang et al. [22] proposed the Adaptive Kalman Filter for the purpose to obtain accurate state estimation of a vehicle's suspension system under different road conditions. Jeong et al. [23] proposed a strategy for relative suspension velocity estimation. The method consists of mathematical modelling and direct measurements provided by an IMU sensor. Vertical front suspension forces are calculated using governing equations combined with heave, roll and pitch accelerations measured by the sensor. Out of these forces, relative suspension velocity is derived. Vazquez et al. [24] provided suspension state, road profile and transfer load estimation methodology using deflection sensors, accelerometers and gyrometers. The observation scheme used is a linear Kalman filter. Despite the approach's simplicity, its robustness against uncertainty is remarkable.

As it can be seen, the most commonly used techniques for suspension state estimation involve Kalman Filter; we propose a new approach in this investigation. The main contribution of this paper investigates the feasibility of a Neural Network (NN) model-based virtual sensor for unsprung mass relative velocity estimation. NNs are increasingly used for sequence regression tasks, including mechanical state prediction [25], lateral vehicle velocity estimation [26,27]). This leads to the assumption that similar methods can be used for virtual sensor development to estimate unsprung mass relative velocity. Based on [26], the Bidirectional Long-short time memory (BiLSTM) based NN model may provide better performance compared to 2 layers of Long-short time memory (LSTM) and 1-dimensional convolutional NN (1D CNN). Therefore, BiLSTM was selected for the NN-model of a proposed virtual sensor for unsprung mass relative velocity estimation in the presented research.

The proposed virtual sensor is designed for real-time application for vehicle unsprung mass relative velocity estimation using multiple sensor data of selected sample window size. Designed virtual sensors decrease the need for physical sensors on vehicle unsprung masses. The output can be used for semi-active/active suspension control. The proposed NN model-based virtual sensor was validated and tested on simulated data, with error data analysis, including time and frequency domains. In addition, error distribution was analysed using an error histogram.

The further structure of the paper is as follows. In Section 2, we present (i) a vehicle model, (ii) manoeuvres used for dataset creation, and (iii) the BiLSTM-based Deep NN model and its optimisation algorithm. Section 3 provides the results of vehicle model validation. Hyperparameter optimisation, BiLSTM-based DNN structure selection is provided in this section as well. Finally, we analyse virtual sensor validation and testing results using the Root Mean Squared Error (RMSE) of unsprung mass relative velocity. In

Section 4, the discussion is performed, and further steps for system development are presented.

2. Materials and Methods

In this Section, materials and methods used for data acquisition and development, validation and testing of the virtual sensor for vehicle relative velocity of unsprung mass are described. First, the vehicle mathematical model is presented, including software descriptions and scenarios. Second, NN-model used for the virtual sensor is described, including model structure and hyperparameter optimisations method.

2.1. Virtual Sensor for vehicle unsprung mass relative velocity estimation

For vehicles equipped with semi-active or active suspensions, the velocities of sprung and unsprung masses are required to implement a control strategy. Sprung mass velocity can be measured using an IMU sensor placed on the sprung mass. The velocity of unsprung mass commonly is evaluated using data from accelerometer and displacement sensors. Our approach for VS creation was supervised learning with selected sensors input and recorded relative unsprung mass velocity data (Figure 1).

The vehicle model of Sport Utility Vehicle (SUV) was built in the IPG CarMaker simulation platform. The model has been parametrised and validated using field tests data from the proving ground; vehicle parameters are presented in Table 1. It is a modified electric Range Rover Evoque vehicle with on-board electric motors. The tire parametrisation using experimental data was performed to simulate tire dynamics.
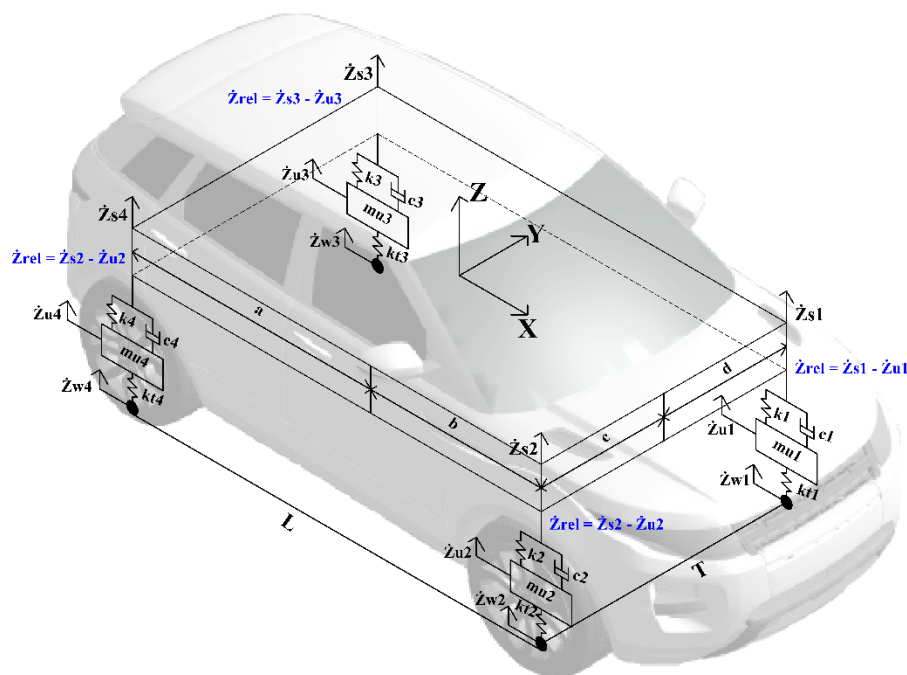

Figure 1. SUV dynamic model

Table 1. Data used in mathematical model

Parameter	Symbol	Value
Wheelbase	L	2.675 m
Distance between front axle and COG	b	1.439 m
Distance between rear axle and COG	a	1.236 m
Height of COG above ground	h	0.65 m
Vehicle mass	m	2442 kg
Total unsprung mass	m_u	126.2 kg
Distance between left track and COG	d	0.778 m
Distance between right track and COG	c	0.847 m
Track width	T	1.625 m
Wheel rotational inertia	J	0.9 kg m²
Tire stiffness	K_t	225368 N/m
Loaded tire radius	R_l	0.343 m
Tire size		235/55/R19
Pitch inertia		642.3 kg m²
Roll inertia		2892 kg m²
Yaw inertia		3231 kg m²

Simulation data included 14 parameters as inputs: sprung mass accelerations in 3 directions (x, y, z), angular rates around these three axes, longitudinal vehicle velocity, front wheels' steering angle and overall vehicle steering angle from steering system, angular velocities of the wheels. The vertical velocities of four unsprung masses were used as an output.

Test tracks and manoeuvres were selected with consideration of data collection for NN training and validation. For training, more dynamic data must be collected; therefore, three scenarios were chosen from the standard scenarios in IPG CarMaker (Figure 2).

DEU Hockenheimring track has an overall length of 2.6 km. There are some straight sections as well as hard cornering. Various speeds up to 120 km/h and driving manners were applied to cover most vehicle dynamics. Secondly, a rural road roundabouts scenario was selected. This scenario contains braking and acceleration manoeuvres. Also, lane changes and driving around the constant radius turns. The overall length driven by the vehicle on this track is 1.76 km, with the maximum speed was 75 km/h.

Furthermore, road DEU Heilbronn was used. This curvy road contains inclinations and declinations. Additional user-defined stopping sections were used to induce acceleration and deceleration manoeuvres. The overall track length covered was 5.79 km with an average speed of 55 km/h. The driver model was modified to three different parameter sets featuring defensive, normal, and aggressive driving styles in all mentioned test scenarios. Different longitudinal, lateral acceleration levels and varying cruising speeds define each driving style.

a)   b)   c)
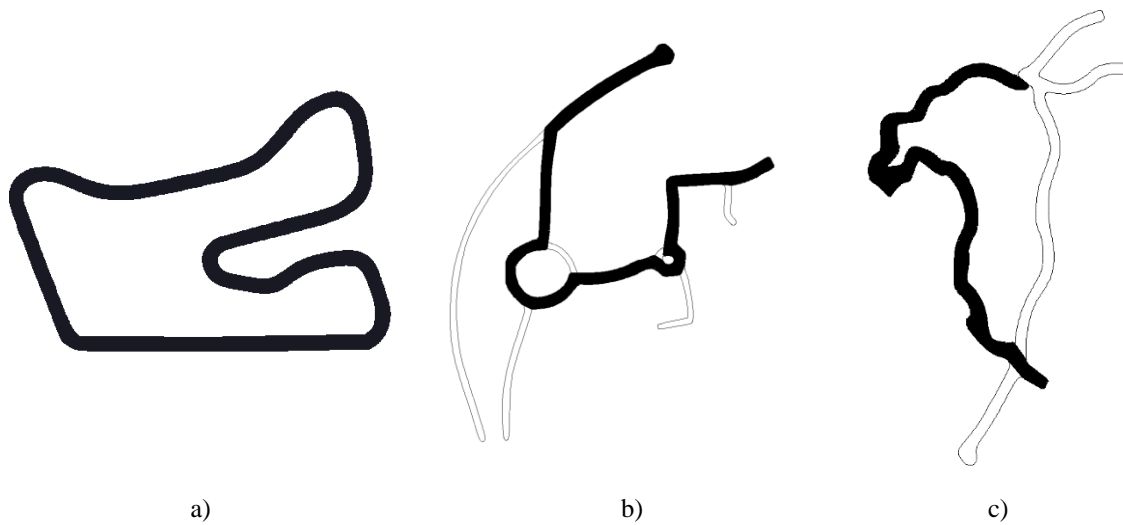
Figure 2. Selected tracks a) Hockenheimring; b) Rural roundabouts road; c) Heilbronn

Data from Heilbronn and rural road tracks were used for training and Hocken-heimring for validation. Additionally, constant radius cornering (ISO 4138:2012, obstacle avoidance, Sine with Dwell (ISO 19365:2016), bumpy road and slalom manoeuvres were simulated for NN dataset creation (see Table 3), and the data was used for NN testing.

2.2. BiLSTM-based Deep Neural Network model

To take advantage of sensor data being sequential, bidirectional long-short term memory (BiLSTM)-based recurrent network was used. This type of neural network (NN) is attributed to recurrent neural networks (RNN) and can learn sequential data models and base prediction on past and current signal values. Model structure, hyperparameter selection experiment and results of validation and testing are presented in this section.

The structure of the selected RNN includes six layers, see Figure 3.
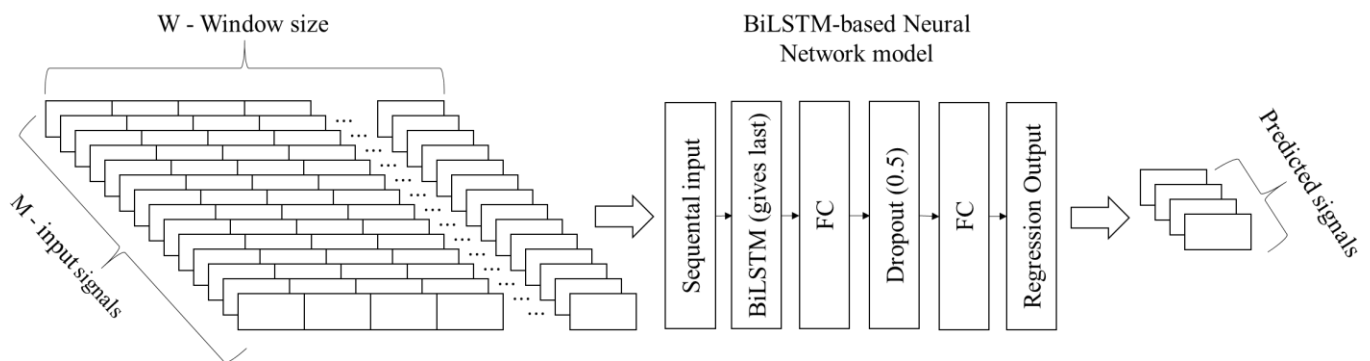

Fig 3. Structure of BiLSTM-based RNN, its inputs and outputs.

NN model layers include sequence input layer, BiLSTM layer, one hidden FC layer, Dropout layer, one output FC layer and regression output. The sequential input layer lets in data from all input channels as a sample sequence of length defined by the window size parameter. BiLSTM layer includes extended LSTM units that propagate signals forward and backwards. That may improve model performance. Hidden FC layer and output FC layers process features extracted in the BiLSTM layer. Units count of the Output FC layer is equal to the outputs count of the virtual sensor. The dropout layer is between the hidden FC and the output FC layer. It reduces NN dependence on single features from BiLSTM, as dropout rate of 0.5 is based.

The input of the network is sequential samples of all inputs M for window size W. The input data samples are fed to the network sequentially, and only the output of BiLSTM for the last input sequence sample is forwarded to the next layer. Therefore, calculation duration in the BiLSTM layer depends linearly on window size W. As each BiLSTM unit does the same calculations, matrix and vector calculation is used. As a result, calculation speed depends less on the overall unit count in NN layers such as BiLSMT and FC. All operations can be done in one cycle of the parallel processing unit and especially graphical processing units (GPU). This network model used in the real-time situation would require holding input data samples that cover selected window size W. The model unit counts are limited to 512 and window size to 51. Therefore, model implementation on low-power devices and automotive on-board computers is feasible.

Even a tiny artificial NN has at least some hyperparameters. They are defined before training and are not optimised during the training process. However, hyperparameters have a significant impact on NN performance and need careful selection or optimisation. As hyperparameters are changed before training, NN training and validation operations are performed for each combination. Therefore, training itself is a long operation, and selecting hyperparameter is times longer. In order to formalise the process of hyperparameter selection, unique optimisation methods are used.

There are three main methods for hyperparameter optimisation: grid search, random search, and Bayesian search. Grid search is the simplest but most computationally expensive. It involves iterating through the defined multi-dimensional grid of hyperparameters combinations. If the step size in this grid is small, iterations count can become huge, and this type of optimisation may take a very long time and many computation resources. It is possible that the grid is too sparse and will not provide the best possible solution. Also, significant time will be spent on unpromising combinations. Random search has no defined grid, and parameter combinations are generated randomly; the random process may be faster compared to grid search if the probability distribution is uniform. Random search has shown that it can find good combinations faster compared to grid search. Bayesian optimisation is similar to random search. Instead of randomly selecting pairs, it analyses previous combinations results, builds the Gaussian probability model, and makes a new combination selection to improve the model (see Figure 4). It saves all combinations, always saving the best, also it provides an estimate that may be not tested yet but provides even better performance. Hence, time is reduced and avoids the drawbacks of grid search.

3. Results

3.1. Vehicle Model Validation and Dataset Generation

The vehicle model has been parametrised and validated on an IPG CarMaker-based simulation platform using field tests data from the proving ground [28]. The obstacle avoidance manoeuvre is presented in Figure 4 as a validation example, achieved accuracy RMSE=0.39 m/s². The test is performed to determine vehicle nature at a severe lane-change manoeuvre. Overall track length is 61 m. All sections should be marked with cones of a minimum height of 500 mm. ISO 3888-2:2011 [29] does not specify a minimum or limit velocity level, but throttle application must be stopped in 2 m after entering section 1.
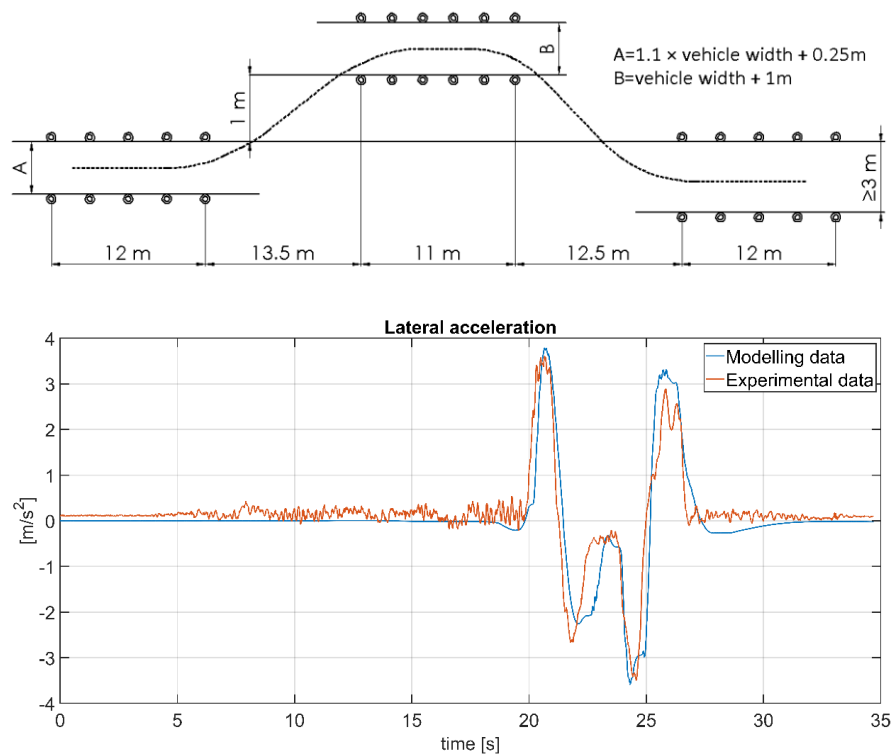
Figure 4. Obstacle avoidance test (ISO 3888-2:2011): a) manoeuvre scheme; b) sprung mass lateral accelerations during the test.

After vehicle model validation, the dataset was generated for NN training, validation and testing. All the roads and manoeuvres are described in Section 2, as well as input and output data.

3.2. Results of Hyperparameter Optimisation

During hyperparameter optimisation of the selected NN model, the predefined window size was used for data samples and selectable unit counts in BiLSTM and FC layers. The window sizes of 3, 7, 11, 17, 19, 21, 25, 31 and 51 have been used. The bigger the window size, the more features can be extracted from the signals, especially the lower frequency and more complex features. On the other hand, a bigger window results in linearly increasing computation duration to get the final output of the sequence from the model. Therefore, a trade-off between the duration of computation and accuracy should be introduced.

During the Bayesian Search, the unit counts in BiLSTM, and FC layers are being selected from range [1, 512]. The models with each selected combination of unit counts are trained and validated on Nvidia Geforce 2080 Ti graphical processing unit (GPU) using MATLAB. For each window size, the mini-batch size is selected to fill the GPU memory as much as possible, as models are pretty small and in cases of small window size do not utilise GPU completely. The total count of training made was around 350. ADAM optimiser used for training with 0.001 learning rate and training was limited to 30 epochs.

RMSE of unsprung mass relative velocity was used as a metric for optimisation. As two parameters were optimised and one metric was used, a 3D mesh can be drawn to show the optimisation process. In Figure 5, hyperparameter optimisation mesh for a window size of 21 is shown.

Training and validation error graphs showed that RMSE reduces mainly in 15 epochs and not much after 25 epochs. There were no overfitting detected in graphs of the training process. The hyperparameter optimisation process that was wrapping the training would reject overfitting networks based on validation RMSE after each training.

In Figure 5, it can be observed that blue dots correspond to tested combinations, black for following possible combination, red star shows feasible model minimum point, and

red mesh shows model mean. Under that mesh, there is a 2D graph with isolines of various colours. Blue points represent low values, and yellow ones correspond to high values of the estimated objective, RMSE. This model helps the search algorithm to select the following points in hyperparameter space. However, the 2D graph of observed and estimated objective function value on the y axis and trial number on the x-axis is provided.
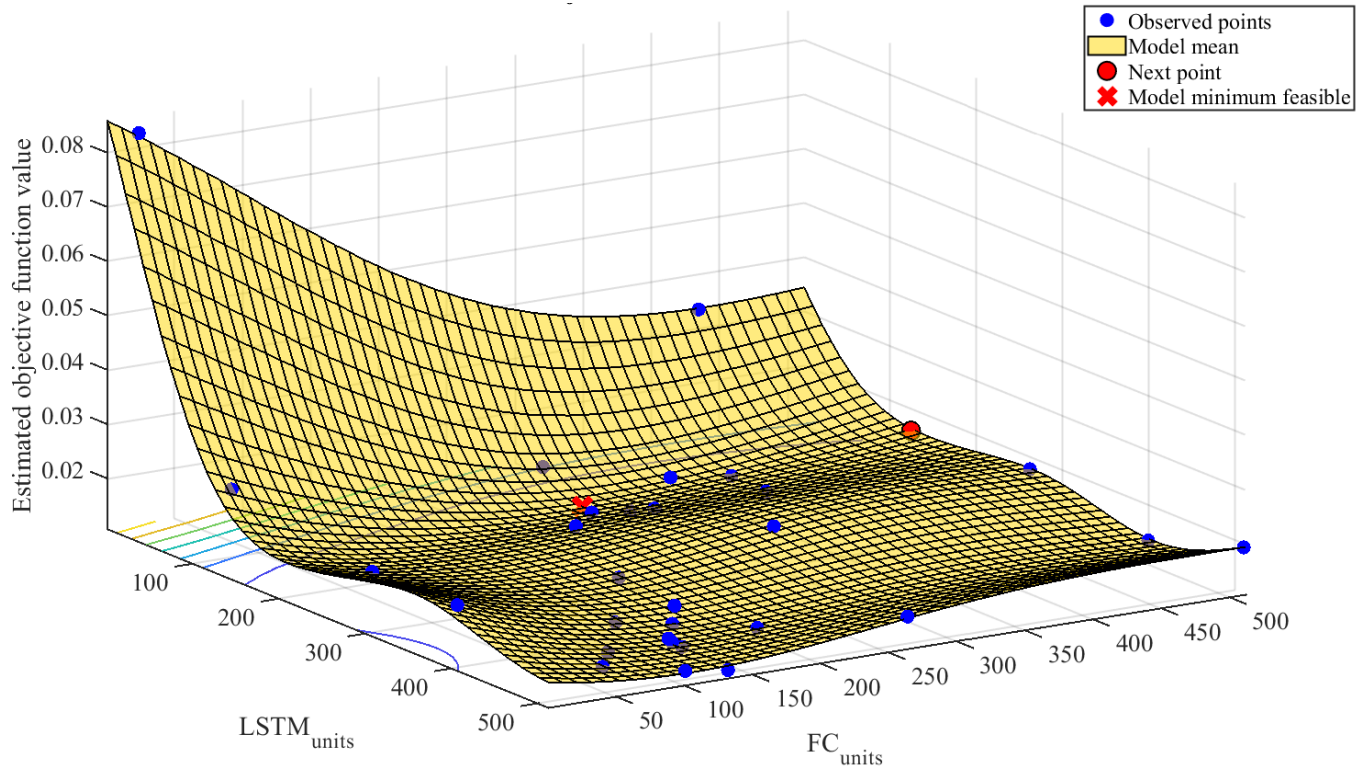

Figure 5. Bayesian search for NN model using sample window of 21.

After a Bayesian search for 30-60 iterations for each window size, only the best network models were selected for each sample window size. The best models are presented with their hyperparameters, RMSE and relative calculation duration in Table 2. The calculation duration is based on sequential calculation (minibatch size of 1) of all validation samples, which are 19,800 minus (window size – 1). As duration depends on processor speed, only relative calculation time is presented compared to the smallest sample window as a baseline, which is about 36 seconds for all samples. The calculation time measurement has been performed three times because of temporal dependencies on hardware performance.

Table 2. Hyperparameter optimisation results with relative performance and calculation duration comparison.

Window size	Selected parameters		RMSE	Relative Error, %	Relative calculation duration, %
	BiLSTM units	FC units			
3	360	403	0.0171	100.0	100.0 (37.4 s)
7	502	295	0.0127	74.3	103.5 (38.7 s)
11	202	312	0.0115	67.3	114.2 (42.7 s)
17	512	111	0.0091	53.2	127.8 (47.8 s)
19	167	256	0.0081	47.4	130.5 (48.8 s)
21	137	298	0.0082	48.0	131.6 (49.2 s)
25	207	511	0.0090	52.6	139.4 (52.1 s)
31	116	345	0.0096	56.1	152.1 (56.9 s)
51	137	298	0.0100	58.5	189.3 (70.8 s)

The results show that RMSE reduces when window sizes from 3 to 19 are used. Therefore, accuracy is being improved, and relative calculation time increases with sample window size growth. The graphs of relative performance and calculation duration are shown in Figure 6. This graph demonstrates that the size of the window of 19 brings the best improvement in performance compared to the increase in calculation duration; longer sample windows bring diminishing returns.
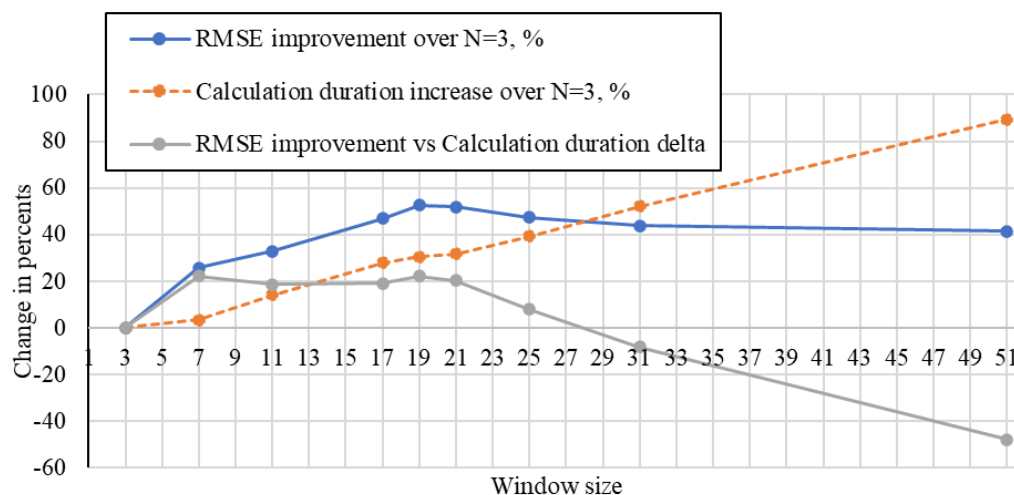
Figure 6. Relative performance and calculation time dependence on sample window size.

In Figure 6, the best RMSE improvement compared to windows size 3 is achieved using window size 19. RMSE improvement in percent and calculation duration delta shows that the most significant positive delta is achieved at a window size of 7 and 19. After evaluating these graphs, the conclusion can be made that the best sample window size is 19. Also, it brings the best compromise between accuracy and computation duration. Therefore, validation and testing results are provided for the window size 19 in the next section.

3.3. Virtual sensor validation and testing

In this section, we provide and analyse virtual sensor outputs on validation and testing data in comparison to reference data, which is unsprung mass relative velocities provided as part of simulated vehicle parameters. The developed virtual sensor estimates unsprung mass relative velocity in the vertical direction. The time series and frequency diagrams of the original and predicted vertical velocity of the unsprung mass is presented in Figure 7.

The time-domain graphs of original simulation output and NN model output for one wheel are provided for validation simulation of Hockenheimring with the normal driving scenario in Figure 7.a.

In order to better present an actual difference between actual and estimated velocity, the corresponding absolute error is shown in Figure 7.b. The max. absolute error is about 0.11 m/s while RMSE is 0.0081. The spikes in absolute error correspond with higher frequency changes in unsprung mass relative velocity and acceleration.

Frequency analysis has been performed to understand how the difference is spread over the spectrum of measurable frequencies. The spectrum relative error was calculated as the delta between predicted and reference signal spectrums divided by the spectrum of the reference signal. The results are shown in Figure 7.c. Measurements are made at 100 samples per second; we provide graphs from 0 to 15 Hz based on primary and secondary ride quality assessment. The observed relative error is mostly between 0 and 10 Hz. The max. relative error is concentrated around 3 Hz. Additionally, in Figure 8, the error histogram shows how predicted signal error values are distributed.
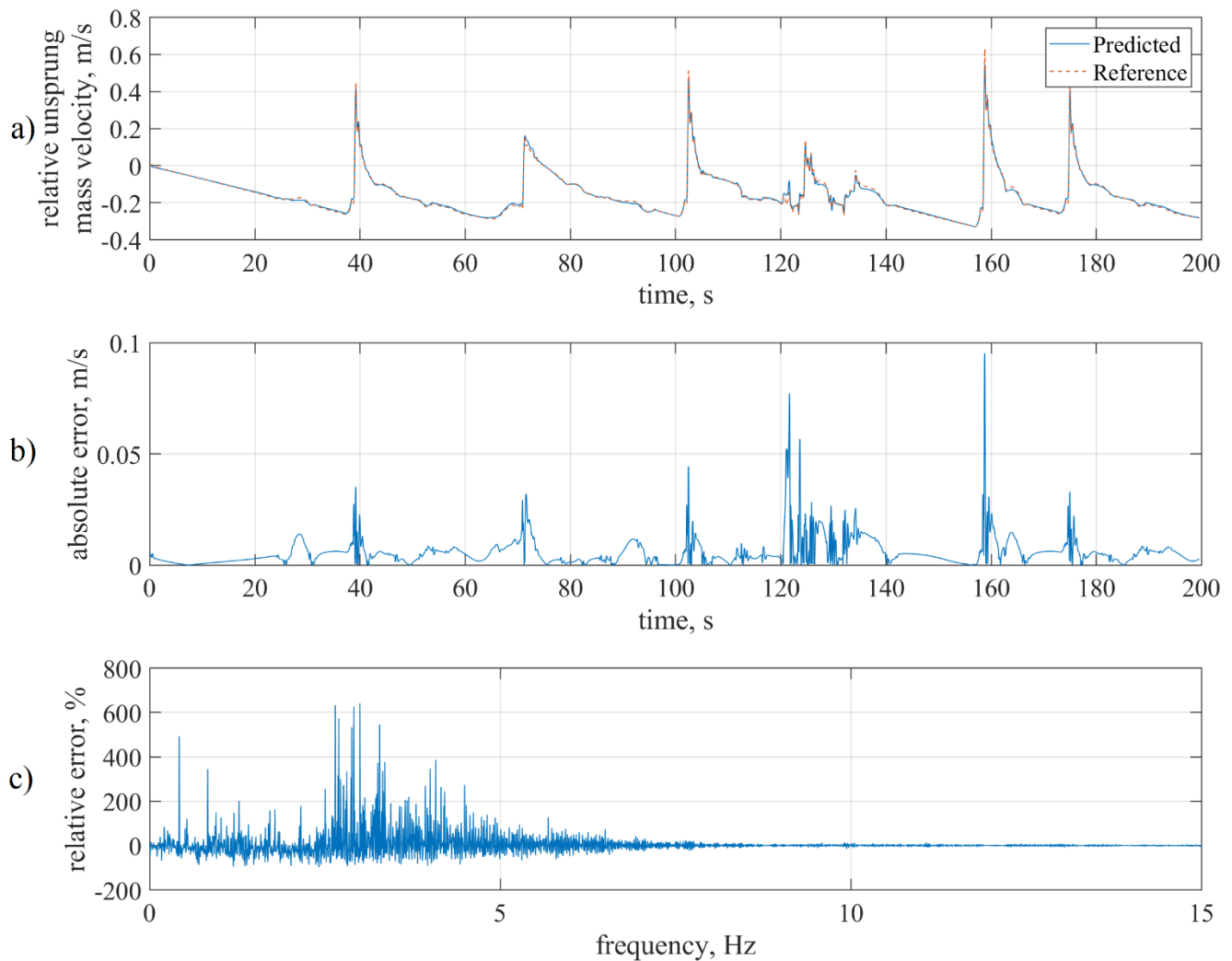
Figure 7. Comparison of predicted and reference signal: a) Time-domain graph of reference and predicted signal; b) b) Time-domain graphs of absolute error of predicted signal compared to reference; c) Relative error of frequency spectrum of a predicted signal.

In addition to Figure 7.b, Figure 8 shows that most errors are between -0.01384 and 0.01402 m/s. A small disbalance around zero is related to the asymmetric bin ranges, considering that the form of distribution meets Gaussian distribution.
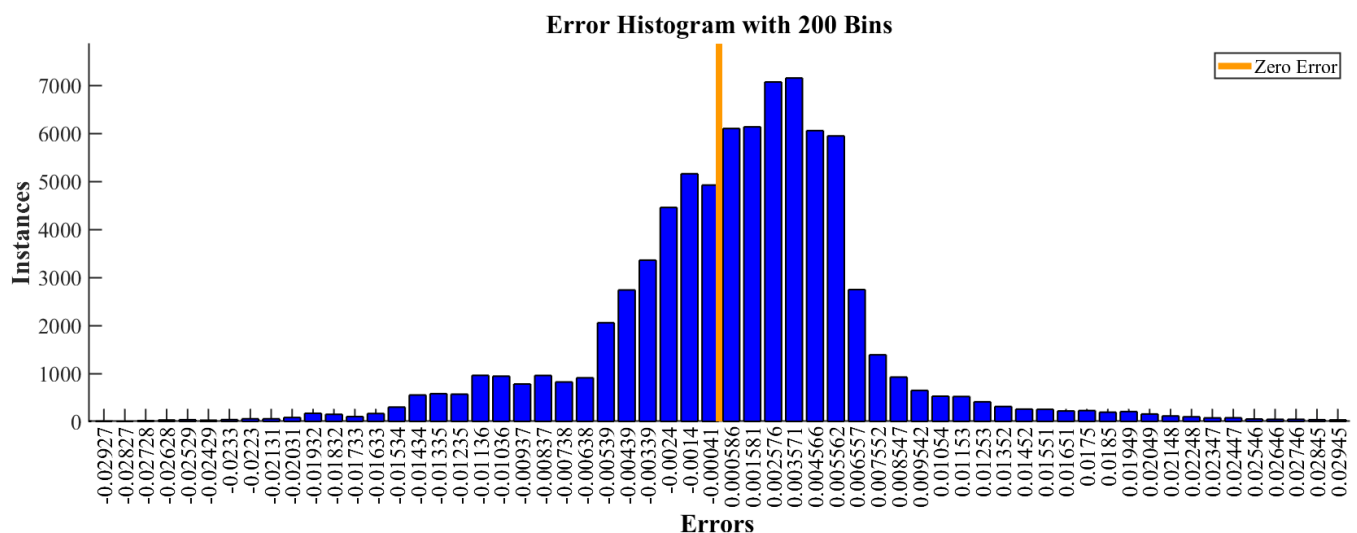
Figure 8. Error histogram for unsprung mass relative velocity prediction compared to the reference.

Complete testing procedure involved 3 validation and 20 testing simulations, and RMSE results calculated for each wheel (unsprung mass) relative velocity prediction separately and overall RMSE of simulation. The results are shown in Table 3. Furthermore, the training RMSE was included to validate the training correctness, as RMSE on the training set should be smaller than on the validation and testing sets. In the data we see, there is not much noise in higher frequency (Figure 7 c). Therefore, the assumption is that augmenting input sensor data with simulated sensor and process noise may improve generalisation.

Table 3. RMSE achieved in various training, validation and testing scenarios for each wheel and overall.

Scenario	RMSE				
	FL	FR	RL	RR	Overall
Heilbronn track, Aggressive driver (training)	0.0034	0.0033	0.0035	0.0034	0.0034
Heilbronn track, Offensive driver (training)	0.0021	0.0020	0.0021	0.0020	0.0021
Heilbronn track, Normal driver (training)	0.0027	0.0026	0.0027	0.0026	0.0027
Rural track, Aggressive driver (training)	0.0068	0.0069	0.0072	0.0071	0.0070
Rural track, Offensive driver (training)	0.0027	0.0028	0.0028	0.0029	0.0028
Rural track, Normal driver (training)	0.0045	0.0046	0.0042	0.0044	0.0044
Hockenheimring track, Aggressive driver (validation)	0.0167	0.0180	0.0157	0.0158	0.0166
Hockenheimring track, Offensive driver (validation)	0.0053	0.0054	0.0046	0.0050	0.0051
Hockenheimring track, Normal driver (validation)	0.0086	0.0091	0.0070	0.0078	0.0081
Constant turn with radius of 100 m at 100 km/h (testing)	0.0393	0.0296	0.0500	0.0500	0.0431
Constant turn with radius of 100 m at 50 km/h (testing)	0.0013	0.0013	0.0013	0.0013	0.0013
Constant turn with radius of 100 m at 75 km/h (testing)	0.0025	0.0036	0.0044	0.0041	0.0037
Constant turn with radius of 30 m at 30 km/h (testing)	0.0008	0.0011	0.0011	0.0011	0.0010
Constant turn with radius of 30 m at 50 km/h (testing)	0.0148	0.0125	0.0189	0.0203	0.0169
Constant turn with radius of 60 m at 50 km/h (testing)	0.0008	0.0010	0.0012	0.0012	0.0011
Constant turn with radius of 60 m at 75 km/h (testing)	0.0313	0.0252	0.0397	0.0411	0.0349
Double lane change (ISO-3888-2) at 30 kmh (testing)	0.0014	0.0013	0.0017	0.0014	0.0015
Sine with Dwell 60 deg at 40 km/h (testing)	0.0029	0.0020	0.0031	0.0022	0.0026
Sine with Dwell 60 deg at 60 km/h (testing)	0.0101	0.0061	0.0108	0.0067	0.0087
Sine with Dwell 60 deg at 80 km/h (testing)	0.0189	0.0133	0.0203	0.0138	0.0169
Sine with Dwell 80 deg at 40 km/h (testing)	0.0037	0.0028	0.0043	0.0025	0.0034
Sine with Dwell 80 deg at 60 km/h (testing)	0.0130	0.0081	0.0145	0.0081	0.0113
Sine with Dwell 60 deg at 80 km/h (testing)	0.0284	0.0226	0.0312	0.0231	0.0266
Bumpy road at 15 km/h (testing)	0.0154	0.0164	0.0209	0.0146	0.0170
Bumpy road at 25 km/h (testing)	0.0223	0.0236	0.0298	0.0209	0.0223
Bumpy road at 32 km/h (testing)	0.0347	0.0359	0.0472	0.0390	0.0395
Slalom 18 m at 15 km/h (testing)	0.0013	0.0014	0.0016	0.0013	0.0014
Slalom 18 m at 25 km/h (testing)	0.0012	0.0013	0.0013	0.0012	0.0012
Slalom 18 m at 35 km/h (testing)	0.0025	0.0017	0.0024	0.0019	0.0021

Based on Table 3, first, aggressive driving increases RMSE, which may be related to higher acceleration and jerk in vehicle motion, as unsprung mass movement depends on road profile and drivers behaviour. Second, higher vehicle speeds increase the RMSE of prediction, as higher speed means higher frequencies, which can be explained by the fact that higher speed increased the frequency of change in unsprung mass relative velocity. Higher frequency signals caused by abrupt manoeuvres may include signal frequencies that are beyond Nyquist frequency for 100 Hz sampling rate that may lead to aliasing in the spectrum.

4. Discussion

This research aimed to develop a virtual sensor for unsprung mass relative velocity prediction based on other vehicle movement characterising sensor data, including IMU and steering related sensors. The hypothesis was that this problem could be solved as a sequential signal regression task. Literature review showed that state-of-art multi-input sequence regression could be implemented using recursive NN, and BiLSTM based models are currently achieving impressive results.

In order to implement an NN-based virtual sensor, training, validation and testing datasets are needed. These datasets were acquired from simulations using IPG Carmaker based on the experimentally validated vehicle model. The simulations covered two tracks and three various driving styles for training, one track and three different artificial driving styles for validation, and five manoeuvres for testing with various parameters and road profiles.

The NN structure has been selected for multi-input sequential regression. The hyperparameter optimisation using Bayesian search was made to select the best parameters. The experiment was done to evaluate the developed models' performance as main metrics RMSE and relative calculation time were selected. Based on the simulations, the NN model with a window size of 19 provides the best performance improvement compared to the computation duration increase.

During the simulation studies, virtual sensor output signals were compared to the reference and analysed in the time and frequency domain. The best NN model demonstrates that the predicted signal is close to reference with an RMSE of 0.0081 and a maximum error of 0.11 m/s. Most errors are concentrated between -0.01384 and 0.01402 m/s, and the error distribution is Gaussian. Frequency domain analysis shows that most of the error is between 0 and 10 Hz, with peak values at 3 Hz.

Analysing the achieved results, it can be concluded that virtual sensors using an NN model such as BiLSTM is a viable option for unsprung mass relative velocity prediction. The final NN consists of 167 BiLSTM, 256 hidden FC units and four output FC units. This virtual sensor running on Nvidia Geforce 2080 Ti GPU, even sequentially (with a mini-batch size of 1), can process inputs at 500 Hz frequency, and where the limitation is data bus latency. In future, the proposed virtual sensor will be assessed on dSPACE real-time target machine, and later on embedded systems that are closer in performance to modern vehicles microcontrollers.

Author Contributions: Conceptualisation, M.D., V.S., E.Š.; validation, P.K., Š.Š. and E.Š.; formal analysis, B.S., V.I., V.S., P.K and E.Š.; data curation, P.K. and E.Š.; software, P.K and E.Š.; visualization, P.K. and E.Š; writing—original draft, all authors. All authors have read and agreed to the published version of the manuscript."

Data Availability Statement: data is available by request from E.Š.

References

1. Campbell, S.; O'Mahony, N.; Krpalcova, L.; Riordan, D.; Walsh, J.; Murphy, A.; Ryan, C. Sensor Technology in Autonomous Vehicles: A review. *In 29th Irish Signals and Systems Conference.* Belfast, UK, **2018**, 1–4, doi: 10.1109/ISSC.2018.8585340.
2. Kocić, J.; Jovičić, N.; Drndarević, V. Sensors and Sensor Fusion in Autonomous Vehicles. *In 26th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, **2018**, 420–425, doi: 10.1109/TELFOR.2018.8612054.
3. Marti, E.; de Miguel, M. A.; Garcia, F.; Perez, J. A Review of Sensor Technologies for Perception in Automated Driving. *IEEE Intelligent Transportation Systems Magazine* **2019**, 11(4), 94–108, doi: 10.1109/MITS.2019.2907630
4. Pinchon, N.; Cassignol, O.; Nicolas, A.; Bernardin, F.; Leduc, P.; Tarel, J. P.; Brémond, P.; Bercier, E.; Brunet, J. All-Weather Vision for Automotive Safety: Which Spectral Band? *Microsystems for Automotive Applications* **2018**, 3–15. Springer, Cham. https://doi.org/10.1007/978-3-319-99762-9_1
5. Skrickij, V.; Šabanovič, E.; Shi, D.; Ricci, S.; Rizzetto, L.; Bureika, G. Visual Measurement System for Wheel–Rail Lateral Position Evaluation. *Sensors* **2021**, *21*, 1297. https://doi.org/10.3390/s21041297
6. Kerst, S.; Shyrokau, B.; Holweg, E. Reconstruction of wheel forces using an intelligent bearing. *SAE Int. J. Passenger Cars-Electronic and Electrical Systems* **2016**, 9(1), 196–203, doi: 10.4271/2016-01-0092.

7. Iyer, K.; Shyrokau, B.; Ivanov, V. Offline and Online Tyre Model Reconstruction by Locally Weighted Projection Regression. In 2020 IEEE 16th International Workshop on Advanced Motion Control (AMC), **2020**, pp. 311-316, doi: 10.1109/AMC44022.2020.9244310.

8. Li, H.; Yu, D.; Braun, J. E. A review of virtual sensing technology and application in building systems. *HVAC&R Research*. **2011**, 17(5), 619–645. doi: 10.1080/10789669.2011.573051

9. Kerst, S.; Shyrokau, B.; Holweg, E. A Model-based approach for the estimation of bearing forces and moments using outer ring deformation. *IEEE Transactions on Industrial Electronics* **2019**, 67(1), 461–470, doi: 10.1109/TIE.2019.2897510.

10. Mattera, C. G.; Quevedo, J.; Escobet, T.; Shaker, H. R.; Jradi, M. A Method for Fault Detection and Diagnostics in Ventilation Units Using Virtual Sensors. *Sensors*. **2018**, 18(11), 3931. https://doi.org/10.3390/s18113931

11. Hensel, B.; Kabitzsch, K. Generator for modular virtual sensors. *In IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA),* Berlin, Germany, **2016**, 1–8. doi: 10.1109/ETFA.2016.7733726.

12. Qin, X.; Zhang, W.; Gao, S.; He, X; Lu, J. Sensor Fault Diagnosis of Autonomous Underwater Vehicle Based on LSTM. *In 2018 37th Chinese Control Conference* (CCC), **2018**, pp. 6067-6072, doi: 10.23919/ChiCC.2018.8483218.

13. Kerst, S.; Shyrokau, B.; Holweg, E. A semi-analytical bearing model considering outer race flexibility for model based bearing load monitoring. *Mechanical Systems and Signal Processing* **2018**, 104, 384–397, doi: 10.1016/j.ymssp.2017.11.008.

14. Acosta, R. M.; Kanarachos, S. Fitzpatrick, M. A Virtual Sensor for Integral Tire Force Estimation using Tire Model-Less Approaches and Adaptive Unscented Kalman Filter. *In International Conference on Informatics in Control, Automatics and Robotics*, **2017**. doi: 10.5220/0006394103860397.

15. Zaharia, C.; Clenci, A. Study on Virtual Sensors and their Automotive Applications. *Scientific Bulletin Automotive series* **2013**, 23. https://automotive.upit.ro/index_files/2013/2013_A_8_.pdf

16. Kahraman, K.; Emirler, T. M.; Centürk, M.; Güvenç, B. A.; Güvenç, L.; Efendioĝlu, B. Estimation of Vehicle Yaw Rate Using a Virtual Sensor with a Speed Scheduled Observer. *IFAC Proceedings Volumes* **2010**, 43(7), 632–637. doi: 10.3182/20100712-3-DE-2013.00043.

17. Skrickij, V.; Savitski, D.; Ivanov, V.; Skačkauskas, P. Investigation of Cavitation Process in Monotube Shock Absorber. *Int.J Automot. Technol.* 19, 801–810 **2018**. https://doi.org/10.1007/s12239-018-0077-1

18. Ricciardi, V.; Ivanov, V.; Dhaens, M.; Vandersmissen, B.; Geraerts, M.; Savitski, D.; Augsburg, K. Ride Blending Control for Electric Vehicles. *World Electr. Veh. J.* **2019**, 10, 36. https://doi.org/10.3390/wevj10020036

19. Nguyen, M. Q.; Canale, M.; Sename, O.; Dugard, L. A Model Predictive Control approach for semi-active suspension control problem of a full car. *IEEE 55th Conference on Decision and Control*, **2016**, 721–726, doi: 10.1109/CDC.2016.7798353.

20. Milanese, M.; Ruiz, F.; Taragna, M. Linear virtual sensors for vertical dynamics of vehicles with controlled suspensions. *European Control Conference* (ECC), **2007**, 1257–1263, doi: 10.23919/ECC.2007.7068662.

21. Pletschen, N.; Badur, P. Nonlinear State Estimation in Suspension Control Based on Takagi-Sugeno Model. *IFAC Proceedings Volumes*, 47(3), **2014**, 11231–11237, ISSN 1474-6670, ISBN 9783902823625, https://doi.org/10.3182/20140824-6-ZA-1003.02500

22. Wang, Z.; Dong, M.; Qin, Y.; Du, Y.; Zhao, F.; Gu, L.(2017) Suspension system state estimation using adaptive Kalman filtering based on road classification, *Vehicle System Dynamics*. **2017**, 55(3), 371–398. doi: 10.1080/00423114.2016.1267374

23. Jeong, K.; Choi, S. B. Vehicle Suspension Relative Velocity Estimation Using a Single 6-D IMU Sensor. *IEEE Transactions on Vehicular Technology* **2019**, 68(8), 7309–7318. doi: 10.1109/TVT.2019.2920876.

24. Vazquez, A.G.A.; Vaseur, C.; Correa-Victorino, A.; Charara, A. Road profile and suspension state estimation boosted with vehicle dynamics conjectures. *IEEE Intelligent Vehicles Symposium* (IV). **2019**, 1809–1815, doi: 10.1109/IVS.2019.8814242.

25. Chen, Z.; Liu, Y.; Liu, S. Mechanical State Prediction Based on LSTM Neural Network. *In Proceedings of the 36th Chinese Control Conference* July 26–28, **2017**, Dialan, China. doi: 10.23919/ChiCC.2017.8027963.

26. Abdelgawad, N. E. A.; El Mahdy, A.; Gomaa, W.; Shoukry, A. Estimating Vehicle Speed on Highway Roads from Smartphone Sensors Using Deep Learning Models. *In 2019 IEEE 31st International Conference on Tools with Artificial Intelligence* (ICTAI), **2019**, Portland, OR, USA, pp. 979–986, doi: 10.1109/ICTAI.2019.00138.

27. Zhang, B.; Zhao, W., Zou, S.; Zhang, H.; Luan, Z. A Reliable Vehicle Lateral Velocity Estimation Methodology based on SBI-LSTM During GPS-outage. *IEEE Sensors Journal.* **2020**. doi: 10.1109/JSEN.2020.3022056.

28. Vaseur, C.; Van Aalst, S. Test Results at Ford Lommel Proving Ground ESR 11 (Version Final) [Data set]. Zenodo. **2019**. http://doi.org/10.5281/zenodo.3263811.

29. ISO 3888-2:2011. Passenger cars - Test track for a severe lane-change manoeuvre - Part 2: Obstacle avoidance manoeuvre. International Organization for Standardization, Geneva, Switzerland. Available online: https://www.iso.org/standard/57253.html (accessed on 20 May 2021).