Receding-Horizon Prediction of Vehicle Velocity Profile Using Deterministic and Stochastic Deep Neural Network Models

Topić, Jakov; Škugor, Branimir; Deur, Joško

Source / Izvornik: **Sustainability, 2022, 14, 10674 - 10694**

Journal article, Published version
Rad u časopisu, Objavljena verzija rada (izdavačev PDF)

https://doi.org/10.3390/su141710674

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:557695

sustainability

MDPI

Article

Receding-Horizon Prediction of Vehicle Velocity Profile Using Deterministic and Stochastic Deep Neural Network Models

Jakov Topić *, Branimir Škugor and Joško Deur

Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, 10002 Zagreb, Croatia
* Correspondence: jakov.topic@fsb.hr; Tel.: +385-1-6168-555

Abstract: The paper firstly proposes a deterministic deep feedforward neural network model aimed at predicting the city bus velocity profile over receding time horizon based on the following inputs: actual vehicle position, actual velocity or short-term history of vehicle velocities, time of day and day of week. A systematic analysis of the influence of different input subsets, history interval length and prediction horizon length is carried out to find an optimal configuration of NN model inputs and hyperparameters. Secondly, a stochastic version of neural network prediction model is proposed, which predicts expectations and standard deviations of velocity patterns over the receding time horizon. The stochastic model prediction accuracy is verified against the recorded test dataset features, as well as by comparing the predicted velocity expectation with the deterministic model prediction and correlating the predicted velocity standard deviation with deterministic model prediction uncertainty metrics. The verification results indicate that: (i) the deterministic model velocity prediction accuracy is characterized by the R^2 score greater than 0.8 for the prediction horizon length of 10 s and remains to be solid (greater than 0.6) for the horizon lengths up to 25 s; (ii) the actual vehicle position and the velocity history are the most significant input features, where the optimal value of history interval length lies in the range from 30 to 50 s; (iii) the stochastic model have only slightly lower accuracy of predicting the velocity expectation along the receding horizon when compared to the deterministic model (the root mean square error is higher by 2.2%), and it outputs consistent standard deviation prediction.

Keywords: velocity prediction; city bus; deep neural networks; stochastic model; experimental verification

Citation: Topić, J.; Škugor, B.; Deur, J. Receding-Horizon Prediction of Vehicle Velocity Profile Using Deterministic and Stochastic Deep Neural Network Models. *Sustainability* **2022**, *14*, 10674. https://doi.org/10.3390/su141710674

Academic Editor: Elżbieta Macioszek

Received: 20 July 2022
Accepted: 23 August 2022
Published: 26 August 2022

1. Introduction

The design of an optimal energy management strategy of a plug-in hybrid electric vehicle (PHEV) is aimed at minimizing the fuel/energy consumption, while providing a high level of driving comfort and long battery life [1,2]. As such, it can benefit from knowing oncoming driving cycle time profiles, including those of vehicle velocity, road slope, ambient conditions, and similar. For instance, global optimization of PHEV control trajectories, used to establish performance benchmark, requires knowledge of the entire driving cycle in advance [3]. In online control strategies, the control trajectory is carried out over a receding time horizon, typically through Model Predictive Control (MPC) [4]. The main driving cycle-related feature relates to velocity time profile [5] because it directly translates into the power requirements for a particular vehicle. There are strong inherent uncertainties in predicting the vehicle velocity, which are associated with the stochasticity of traffic environment (due to traffic congestion, traffic lights and signs, pedestrians crossing the road, etc.), and which make the formulation and execution of prediction task challenging [6–8]. To summarize, recent research efforts in the field of PHEVs, and in general intelligent vehicles, have been focused on the development of advanced predictive vehicle control strategies, which require accurate and at least short-term predictions of future vehicle velocity profiles [9].

1.1. Literature Review

Due to the stochastic nature of driving cycles, it is difficult to perform accurate vehicle velocity prediction using traditional time series prediction approaches based on the linear regression models, such as ARIMA model [9]. Therefore, more sophisticated models have commonly been used in literature to predict the vehicle velocity, with emphasis on Markov chains and artificial neural networks (NN) [10]. In [8], the authors use the vehicle geographical coordinates, velocity, departure date and time, and historical data from a traffic database as inputs to a long short-term memory (LSTM) NN that predicts velocity on a particular road segment. Despite a rich set of input variables, a relatively rough prediction accuracy was obtained due to the insufficient amount of data used for prediction model training. A somewhat different approach is employed in [11], where an advanced Evolutionary Least Learning Machine (E-LLM) tool is proposed to predict the velocity sequence over a given time horizon, and where an exceptional prediction accuracy is reported, but for a very limited set of test-driving cycles. In [12], the authors systematically compare the performance of different prediction models divided in two main categories: (i) deterministic models that predict velocity expectation, and (ii) stochastic models that predict the entire velocity distribution. Numerous input data are used therein to parameterize prediction models, including vehicle internal data (CAN bus data) and external data, such as traffic light positions and stop signs and upcoming bends (on-board radar and traffic database). In [13], different designs of multivariate Markov chains for vehicle velocity prediction are systematically compared, and an approach based on multiple transition probability matrices (TPM) is proposed, where each TPM^t serves to predict the vehicle velocity t steps away in the future. The main advantage of the NN approach is the ability to automatically learn complex underlying patterns, i.e., non-linear relationships between input features, from the available data [9]. In addition, the NN approach has no limit on the number of input features as opposed to the alternative models based on Markov chains, where the TPM size increases exponentially with the number of input features.

In some studies, vehicle velocity prediction is performed in the position domain [14,15], since uncertainties related to road geometry (bends), traffic lights, bus stops, etc., are directly associated with position rather than time domain. In these approaches, different prediction models are typically combined for better prediction accuracy. For example, in [14] a Markov chain is combined with a NN model, and it is claimed that the combined model improves the prediction accuracy by approximately 25% on average when compared to the individual prediction models. In [15], the velocity profiles are predicted by using a stochastic Markov chain-based model, with vehicle velocity limits determined by applying statistical methods to recorded driving data along the considered route. The main disadvantage of position domain prediction comes from the fact that MPC systems require time profiles of predicted velocity, whereby the process of transforming profiles from the position domain to the time domain is not ambiguous, especially at the zero-velocity singularity point.

The velocity prediction models are typically tested in terms of prediction accuracy, execution time, and predictive control performance [9,10]. For example, in [16] the PHEV fuel consumption reduction performance is assessed for the following three cases: (i) full driving cycle knowledge, (ii) perfect velocity predictions on a receding horizon of 10 s, and (iii) actual velocity predictions on a receding horizon of 10 s obtained by applying LSTM NN model. The common metrics, such as root mean square error (RMSE), mean absolute error (MAE), and a coefficient of determination, i.e., R^2 score between the actual and predicted velocity profiles are commonly used as prediction accuracy indicators.

1.2. Problem Statement

The design of PHEV powertrain configuration and associated control strategy can be more effective if the driving cycles utilized are more realistic and/or if the driving route is known in advance and/or if the statistical features of driving cycles are known from historical traffic data [17,18]. A priori knowledge of driving route and a rich set of historical data are regularly available in delivery and public transport systems due to

known and often repeating driving schedules [19]. This makes the delivery and public transport vehicles suitable for the application of predictive control strategies [20]. To make the control strategy effective and viable, the velocity prediction models should be accurate, robust, and computationally efficient [21].

In the authors' recent conference publication [22], a NN-based static stochastic model (labelled as NN-STC; see Figure 1a) has been proposed for predicting vehicle velocity distribution along a regular city bus route. The NN-STC relies on the vehicle position d_k (i.e., the travelled distance from the departure station) as the only input feature and is capable of predicting the velocity probability distribution at different positions, $\mathbb{P}(\tilde{v}_j \mid d_k)$, based on historical city bus tracking data. Although the analysis of NN-STC prediction accuracy have pointed to solid model performance in terms of capturing the recorded velocity distribution expectation and deviation (Figure 1b), it is evident that the single-input model is not capable to accurately predict individual, actual, and future velocity values. As such, the NN-STC model is not suitable for MPC applications. To overcome this weakness, the model should be extended with richer traffic information in terms of taking into account a wider set of model inputs and history of inputs, and formulating the output as a deterministic or stochastic velocity time profile over receding horizon, which is the main topic of this paper.

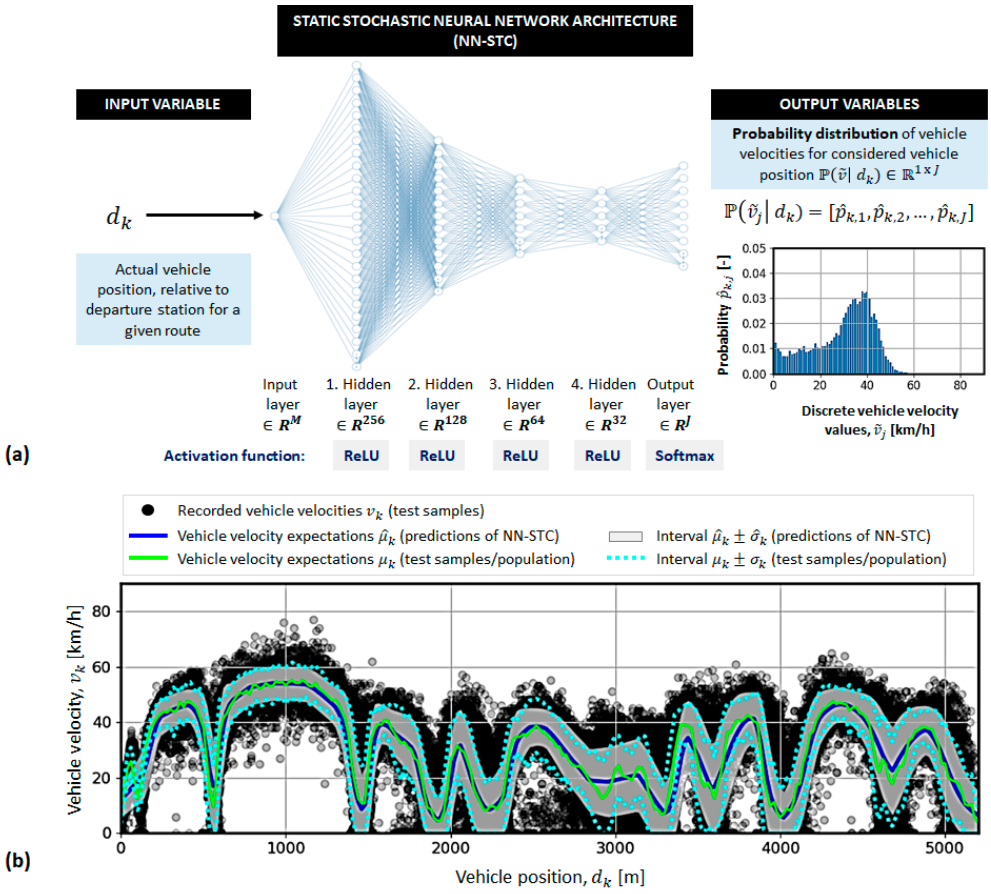

Figure 1. Illustration of previously proposed static stochastic neural network (NN-STC) model [22] for predicting city bus velocity distribution along route (**a**), including related testing results (**b**).

1.3. Research Aim and Contributions

Based on the above state-of-the-art review, it may be concluded that the existing approaches to vehicle velocity prediction typically demands a multitude of input data, such as on-board radar data and traffic database [9,12], which are often unavailable in vehicles. Additionally, despite a rich set of input features used, the testing results of

prediction models are mostly given only for a very limited set of test-driving cycles [8,11]. Finally, the prediction models are typically designed for the case of a generic route, thus relying only on the vehicle velocity and acceleration states as the main driving cycle-related input features [10,13]. As such, they do not consider the vehicle position state as an additional input feature that may significantly contribute to the vehicle velocity prediction accuracy [15], particularly for the vehicles operating over prescribed routes (such as city buses considered herein).

To this end, the paper proposes and systematically tests deterministic and stochastic NN models for predicting city bus velocity time profiles on receding horizon based on readily available vehicle data, which include actual vehicle position, actual vehicle velocity or velocity history sequence, time of the day, and day of the week, while the outputs contain sequence of vehicle velocities over the receding time horizon, including the velocity standard deviation in the case of a stochastic model.

The main contributions of this paper include: (i) design and comprehensive assessment of deterministic and stochastic variants of data-driven model for predicting vehicle velocity on receding time horizon, which is suitable for application in various intelligent transport systems based on the vehicle data only; and (ii) approach to in-depth analysis of the proposed models' prediction accuracy aimed at finding the most suitable model configurations and hyperparameters in terms of selecting relevant input features and setting the velocity history interval and prediction horizon lengths.

The remaining part of the paper is organized as follows. Section 2 describes the overall procedure of city bus driving cycle recording and pre-processing for the need of training, validation, and testing of the proposed vehicle velocity prediction models. Section 3 presents the deterministic velocity prediction model, including a comprehensive analysis of prediction accuracy with respect to different sets of model hyperparameters. Section 4 introduces the stochastic velocity prediction model and presents a comparative analysis of its expectation prediction accuracy with respect to deterministic model. Concluding remarks are given in Section 5.

2. Description of the Recorded Dataset

2.1. Recorded Driving Cycles

The driving cycles have been collected on a set of 10 city buses of the same type (MAN Lion's City NL 323) operating in the city of Dubrovnik [23]. Data recording was performed continuously for a period of six months by using a commercial GPS/GPRS vehicle tracking device installed on selected buses. The sampling time was set to one second to ensure sufficient data density with respect to the dynamics of vehicle velocity change. The relevant GPS recorded data include:

- Bus garage number;
- Timestamp;
- Bus geographical coordinates;
- Bus longitudinal velocity;
- Cumulative distance travelled (from odometer).

The data related to circular route Babin kuk–Pile–Babin kuk are considered herein, because this route has a relatively long length and stretches over different parts of the city. The recorded driving data have been segmented into driving cycles, where each driving cycle is defined by a time series of vehicle velocity and distance travelled between the two end stations, here Babin kuk and Pile. The driving cycles were filtered primarily to select those that do not include GPS signal losses or outliers. The criteria that a driving cycle needed to meet to be declared valid included:

(1) The driving cycle follows the reference route trajectory, i.e., no detours are acceptable;
(2) The time difference between each pair of consecutive driving cycle samples is equal to the nominal sampling time, i.e., 1 s;
(3) The initial and final velocity of the driving cycle is equal to zero;

(4) The vehicle acceleration values are within the interval of [–3, 3] m/s^2;
(5) The total number of numerically undetermined values (NaN) is zero;
(6) The proportion of the dwell time samples to the total number of samples is less than 75%.

This filtration procedure has resulted in the total number of 2313 (91.6%) valid recorded driving cycles.

Figure 2 shows ten randomly selected valid driving cycles in the time and vehicle position domains. The variability of velocity time profiles is due to variations in traffic conditions along the day (Figure 2a). The velocity profiles are more predictable when expressed in the vehicle position domain (Figure 2b) because characteristic road features (e.g., traffic light, bus stops, open road segments, etc.) are spatially determined.

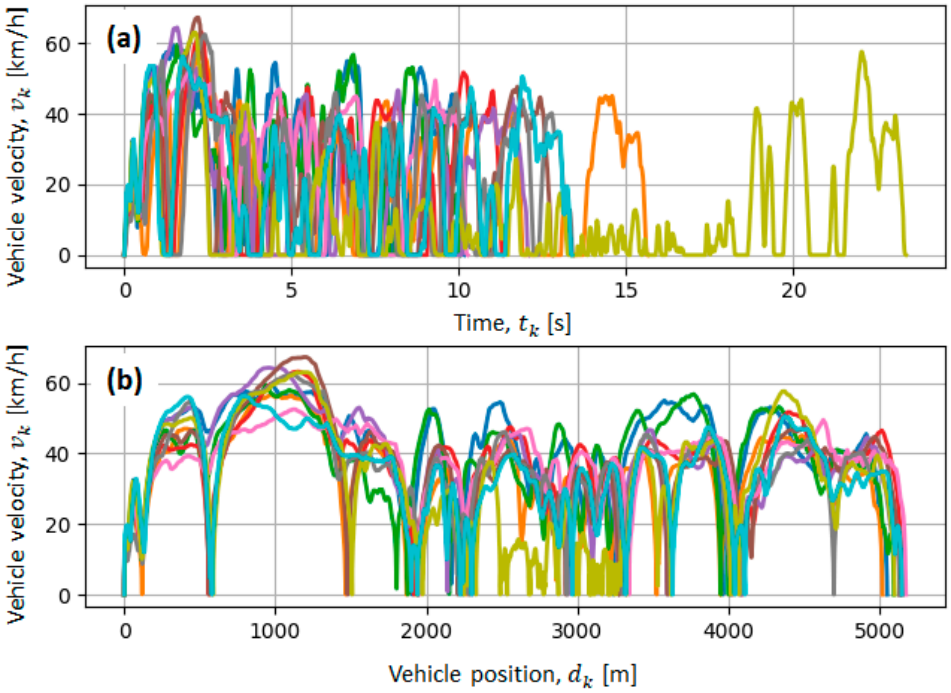

Figure 2. Examples of ten randomly selected valid recorded driving cycles for the direction Babin kuk–Pile: velocity vs. time (**a**) and velocity vs. travelled distance (**b**) profiles.

2.2. Preparation of Training, Validation, and Test Datasets

The full set of 2313 valid driving cycles consists of a total set of 1,908,209 recorded data samples/points. Due to the complexity of the prediction task, learning the NN model on such a wide set of recorded data (≈2 million samples) can be a computationally demanding in terms of NN training time. Therefore, the following method of isolating a reduced set of data samples has been applied [22]: by passing through every 5th consecutive recorded time step $t_{i,k}; k = 1,\ 6,\ 11, \ldots, K$ for every i-th recorded driving cycle, a single sample is randomly selected from each consecutive time interval $T_{i,k} \in [t_{i,k}, t_{i,k+5}]$ and stored in the set $\mathcal{U} = \{u_1, u_2, \ldots, u_n\}$. As shown in Figure 3, each selected time sample t_k represents the reference point r_k from which the H_h history and H_p future discrete velocity values are taken with the original sampling time of 1 s (vectors $v_{h,k}$ and $v_{p,k}$, respectively), including the data on vehicle position d_k, time of day t_k, and day of week w_k.

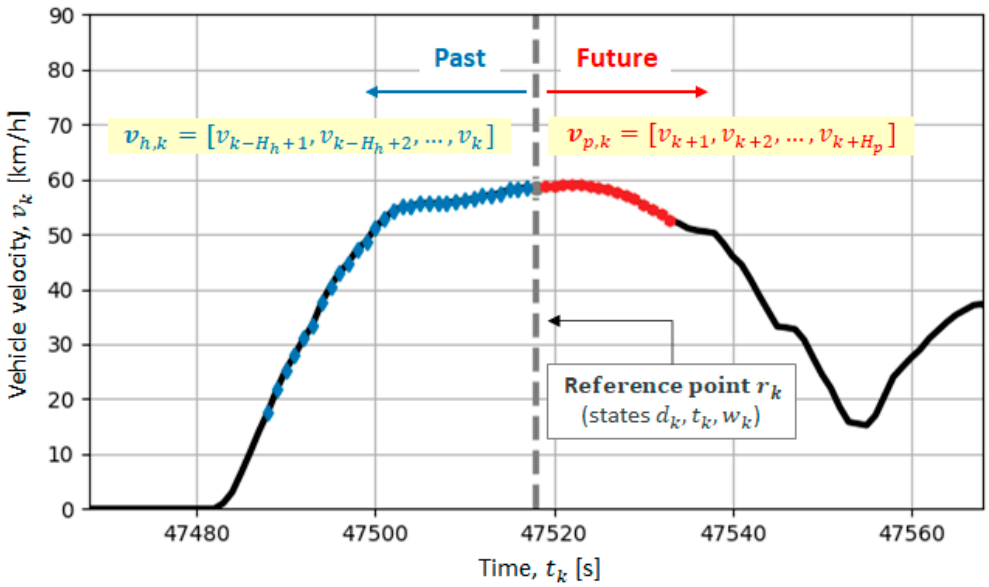

Figure 3. Illustration of the process of generating input-output datasets $\mathcal{X}$ and $\mathcal{Y}$, necessary for training, validation, and testing of NN-based velocity prediction models.

The above pre-processing procedure results in the final/reduced set of 382,553 data samples, which forms NN input dataset $\mathcal{X} \in \mathbb{R}^{382,553 \times (H_h+3)}$ and target values $\mathcal{Y} \in \mathbb{R}^{382,553 \times H_p}$. The datasets $\mathcal{X}$ and $\mathcal{Y}$. are scaled to the interval [0, 1] according to the principle of min–max normalization and randomly divided into sets for NN model training (70% of samples), validation (15% of samples), and testing (15% of samples). The training dataset is used for the initial training/parameterization of the proposed models. The validation dataset is used for tuning the model's hyperparameters and preventing of model overfitting. The test dataset is used to provide an unbiased evaluation of the final model on the previously unseen data.

3. Deterministic Vehicle Velocity Prediction Model

3.1. Modelling of Deep Neural Network with Deterministic Output

The deterministic velocity prediction model, hereinafter referred to as NN-DET, is based on the deep feedforward NN, whose architecture is shown in Figure 4. Inputs to the NN-DET model include data on the current vehicle position (d_k), the current vehicle speed v_k (or velocity history $v_{h,k}$, as an alternative), the time of day t_k, and the day of week w_k. The output from the NN-DET model is the vector of predicted vehicle velocities $\hat{v}_k = \left[\hat{v}_{k+j}\right]; j = 1, 2, \ldots, H_p$, on the receding time horizon of length H_p, where the index k denotes the discrete time step of the input state $X_k = \{d_k, v_k, t_k, w_k\}$ (see Section 2). The NN-DET model consists of an input layer that receives $M = 4$ input variables/features, four fully connected hidden layers containing 256, 128, 64, and 32 hidden neurons, and an output layer of H_p elements. Note that if the vehicle velocity history vector $v_{h,k} = \left[v_{h,k-H_h+\widetilde{j}}\right], \widetilde{j} = 1, 2, \ldots, H_h$, is used as the second input to NN-DET instead of the current velocity v_k, then the number of inputs of NN-DET model increases by $H_h - 1$, which, consequently, increases the model complexity in terms of total number of adjustable model parameters. The hidden layers use the ReLU activation function, while the output layer contains the linear activation function [24]. The discrete time window lengths H_h and H_p represent hyperparameters, whose optimal values need to be determined through separate analysis.

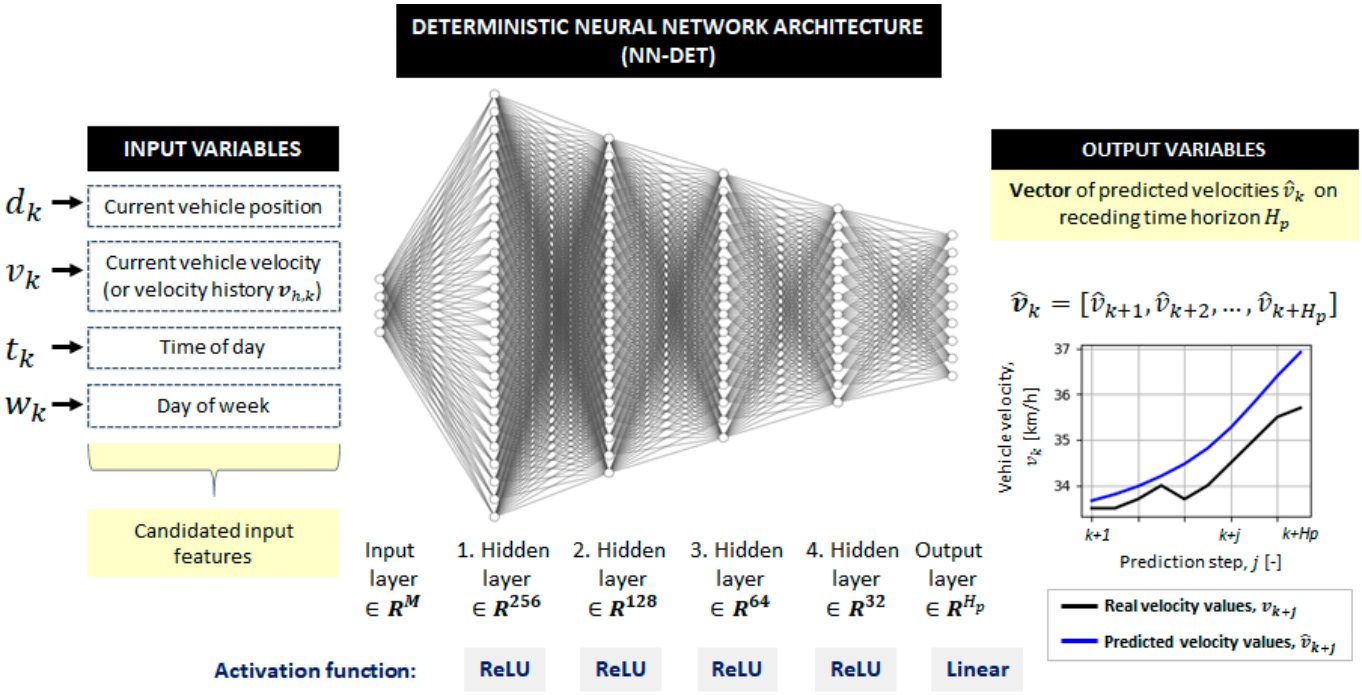

Figure 4. Proposed architecture of deterministic deep feedforward neural network (NN-DET) for predicting vehicle velocity profile on receding time horizon $\hat{v}_{k+j}; j = 1, 2, \ldots, H_p$.

To solve the regression problem, the mean squared error (MSE) between predicted $(\hat{v}_k)$ and actual/recorded (v_k) velocity vectors on the discrete time horizon H_p has been chosen as the loss function to be minimized [24]:

$$MSE_v = \frac{1}{N} \sum_{k=1}^{N} \sum_{j=1}^{H_p} \left(\hat{v}_{k,j} - v_{k,j} \right)^2, \quad (1)$$

where N denotes the total number of samples used to train the NN-DET model (here 382,553; see Section 2). For better generalization of the NN-DET model, each hidden layer is regularized by L_2 regularization method (also known as ridge regression) [24], which adds the "squared magnitude" of the NN coefficients/weights, $\lambda \parallel W \parallel_2^2$, as the penalty term to the loss function MSE_v, with the regularization factor λ set to $5 \cdot 10^{-4}$.

The realization of the NN-DET model from Figure 4 is performed within the Python programming environment by using the *Keras* [25] module with the *Tensorflow* [26] module as the backend. The built-in ADAM optimization algorithm with default settings is used to train the NN-DET model, since it has proven to be particularly suitable for training of deep NNs [27]. The number of samples from which the gradient is calculated (i.e., batch size) is set to 16, while the number of training epochs is set to 150.

3.2. Analysis of Deterministic Model Prediction Accuracy

3.2.1. Influence of Input Features

The main goal of this analysis is to determine the relevant input features for further consideration of NN-DET model. It is of particular interest to assess the benefits of including the current city bus position as an additional input in addition to velocity data that are typically used in the literature (see, e.g., [9]). To this end, the following combinations of input features are considered:

(1) Current vehicle position d_k;
(2) Current vehicle position d_k and current vehicle velocity v_k;
(3) Vehicle velocity history $v_{h,k}$;
(4) Current vehicle position d_k and vehicle velocity history $v_{h,k}$;

(5) Current vehicle position d_k, vehicle velocity history $v_{h,k}$, and time of day t_k;
(6) Current vehicle position d_k, vehicle velocity history $v_{h,k}$, time of day t_k, and day of week w_k.

For each of the above combinations of input features, training of separate NN-DET models is performed, which are denoted as NN-DET-I_n, where n denotes the ordinal number of the input combination as listed above. Note that the fixed prediction horizon length $H_p = 10$ s (and the fixed velocity history interval length $H_h = 20$ s, when applicable) are applied to each model to make the comparative analysis consistent.

The NN-DET prediction accuracy is evaluated based on the root value of the MSE_v loss function defined by Equation (1) and calculated for the case of validation dataset:

$$RMSE_v = \sqrt{MSE_v} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\sum_{j=1}^{H_p}\left(\hat{v}_{k,j} - v_{k,j}\right)^2}. \quad (2)$$

$RMSE_v$ is given in km/h, thus providing better interpretability of results than MSE_v that is given in km^2/h^2.

Figure 5 shows the loss values $RMSE_v$ for each NN-DET-I_n model, which are also given in Table 1 along the number of adjustable NN parameters. The vehicle velocity profile prediction accuracy saturates in the case of NN-DET-I_4 model ($RMSE_v = 5.07$ km/h), which clearly indicates that both time of day t_k and day of week w_k have a negligible influence on the model prediction accuracy. The worst prediction accuracy is obtained for the NN-DET-I_1 model ($RMSE_v = 11.84$ km/h), which relies on the current vehicle position d_k as the only input feature. The reason for this is the insufficient conditionality of the model with respect to the input data, as previously confirmed when predicting individual vehicle velocity values in the case of NN-STC model (Figure 1, [22]). By adding the current velocity v_k next to the vehicle position d_k (NN-DET-I_2 model), a significant but still insufficient drop in $RMSE_v$ from 11.84 km/h to 7.28 km/h (≈40%) is achieved. Despite using the vehicle velocity history $v_{h,k}$ as only input, the NN-DET-I_3 model achieves the prediction accuracy close to the NN-DET-I_4 model ($RMSE_v = 6.12$ km/h vs. 5.07 km/h), which additionally uses the vehicle position input. It is important to note that none of the input feature combinations result in model overfitting, as evidenced by the absence of a gap between training and validation losses (Figure 5). Good model generalization properties are additionally confirmed through the closeness of the testing and validation losses (Table 1).

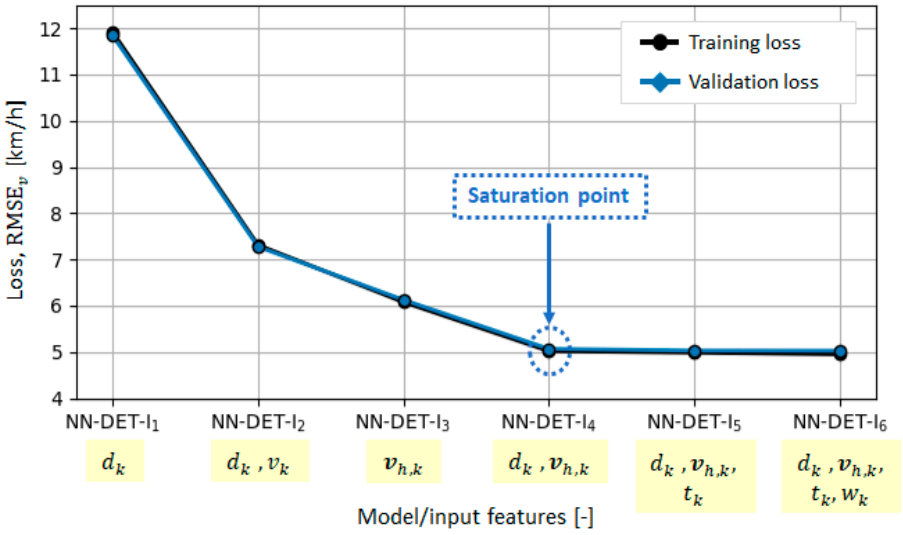

Figure 5. $RMSE_v$ loss values obtained for each NN-DET-I_n model (defined by characteristic input feature sets) by using training and validation datasets.

Table 1. Training, validation, and testing results for each NN-DET-I_n models (defined by characteristic input feature sets), including the number of adjustable model parameters.

Model	Loss Value, $RMSE_v$ [km/h]			Number of NN Parameters
	Training Set	Validation Set	Test Set	
NN-DET-I_1	11.84	11.84	11.83	44,074
NN-DET-I_2	7.26	7.28	7.26	44,330
NN-DET-I_3	6.01	6.12	6.11	48,938
NN-DET-I_4	4.96	5.07	5.07	49,194
NN-DET-I_5	4.92	5.03	5.05	49,450
NN-DET-I_6	4.88	5.03	5.03	49,706

3.2.2. Influence of Vehicle Velocity History Interval Length

To determine the optimal value of the vehicle velocity history interval length H_h, the prediction accuracy of the NN-DET model is examined for the lengths $H_h \in \{3, 7, 10, 15, 20, 30, 40, 50, 100, 200\}$ s. The resulting models are referred to as NN-DET-$H_{h,T}$, where T denotes the history interval length in seconds. Again, the $RMSE_v$ loss values, defined by Equation (2) and calculated for the case of the validation dataset, are used to evaluate the prediction accuracy of the NN-DET-$H_{h,T}$ model. When training each of the NN-DET-$H_{h,T}$ models, the full set of candidate input features (d_k, $v_{h,k}$, t_k, and w_k) is used along the fixed prediction horizon length $H_p = 10$ s.

Figure 6 shows the $RMSE_v$ values for each NN-DET-$H_{h,T}$ model, with the numerical values contained in Table 2. These results point out that the validation loss is locally saturated at $H_h = 50$ s, where $RMSE_v$ equals 4.85 km/h, while in the case of longer velocity history ($H_h = 200$ s) $RMSE_v$ increases to 4.96 km/h. The reason for this is that the NN-DET-$H_{h,200}$ model is overly conditioned by the input data, which leads to model overfitting, i.e., poorer model generalization properties. This is reflected in the increasing gap between the training and validation loss plots in Figure 6, while the validation and testing loss values are close to each other (Table 2). The values of H_h in the range of 30 to 50 s represent an optimal selection, as they provide minimal loss on validation (and test) dataset and reduce the complexity of NN-DET model.

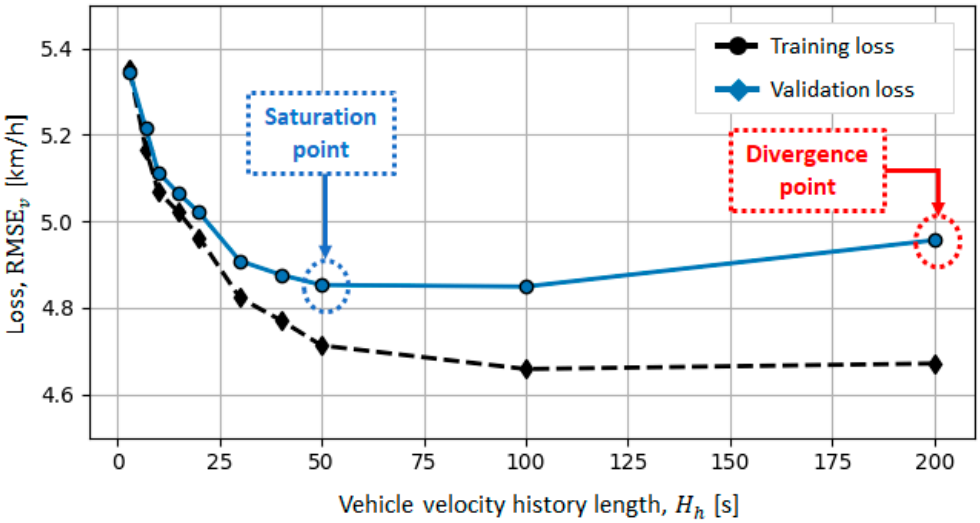

Figure 6. $RMSE_v$ loss values obtained for different velocity history interval lengths H_h of NN-DET model by using training and validation datasets.

Table 2. Training, validation, and testing results of NN-DET models with different history interval length H_h [s], including the number of adjustable model parameters.

Model	Loss Value, $RMSE_v$ [km/h]			Number of NN Parameters
	Training Set	Validation Set	Test Set	
NN-DET-$H_{h,3}$	5.32	5.35	5.38	45,354
NN-DET-$H_{h,7}$	5.13	5.21	5.23	46,378
NN-DET-$H_{h,10}$	5.00	5.11	5.10	47,146
NN-DET-$H_{h,15}$	4.93	5.06	5.05	48,426
NN-DET-$H_{h,20}$	4.91	5.02	5.01	49,706
NN-DET-$H_{h,30}$	4.76	4.91	4.92	52,266
NN-DET-$H_{h,40}$	4.70	4.88	4.86	54,826
NN-DET-$H_{h,50}$	4.61	4.85	4.84	57,386
NN-DET-$H_{h,100}$	4.57	4.85	4.89	70,186
NN-DET-$H_{h,200}$	4.53	4.96	4.93	95,786

3.2.3. Influence of Vehicle Velocity Prediction Horizon Length

The prediction accuracy of the individual vehicle velocity profiles $\hat{v}_k \in \mathbb{R}^{H_p}$ is evaluated for the following receding horizon lengths $H_p \in \{3, 7, 10, 15, 20, 50, 100, 200\}$ s. The prediction models are designated as NN-DET-$H_{p,T}$, where T denotes the receding horizon length in seconds. Each model is trained separately for the following input sets:

a. Full set of candidate features (d_k, t_k, w_k i $v_{h,k}$);
b. Vehicle velocity history only ($v_{h,k}$),

With a fixed velocity history interval length $H_h = 20$ s. Evaluation of the NN-DET-$H_{p,T}$ model prediction accuracy is performed based on the following indicators calculated for the validation dataset:

i. The loss value $RMSE_v$ defined by Equation (2);
ii. The R_j^2 score value, calculated for each jth discrete step of the prediction horizon length H_p:

$$R_j^2 = 1 - \frac{\sum_{k=1}^{N}\left(v_{k,j} - \hat{v}_{k,j}\right)^2}{\sum_{k=1}^{N}\left(v_{k,j} - \overline{v}_j\right)^2}, \quad j = 1, 2, \ldots, H_p. \tag{3}$$

Figure 7 and Table 3 show the loss values $RMSE_v$ for the various prediction horizon lengths H_p, where the indices $_a$ and $_b$ correspond to the above-defined model input sets. These results indicate that the validation loss assumes almost linear upward trend until $H_p = 20$ s for both input set cases, and then begin to saturate to the levels $RMSE_v = 13.22$ km/h in the case of NN-DET-$H_{p,a}$ model (full set of inputs) and $RMSE_v = 15.84$ km/h in the case of NN-DET-$H_{p,b}$ model (only the velocity history input), which are reached at $H_p = 200$ s. The loss $RMSE_v$ is higher by 23% on average for NN-DET-$H_{p,b}$ model when compared to full model NN-DET-$H_{p,a}$, reaffirming the importance of including the current city bus position as an additional input feature of the model.

According to the R_j^2 score results shown in Figure 8, the full, NN-DET-$H_{p,a,200}$ model maintains a satisfactory prediction accuracy ($R_j^2 > 0.25$, for all $j = 1, 2, \ldots,$ H_p, $H_p = 200$) which is not the case with the NN-DET-$H_{200,b}$ model, thus confirming once again the importance of including the current vehicle position d_k in the input feature set. The prediction accuracy is very good ($R_j^2 \geq 0.8$) for the first 10 prediction steps ($j \leq 10$) and remains to be solid ($R_j^2 > 0.6$) up to first 25 prediction steps ($j \leq 25$).

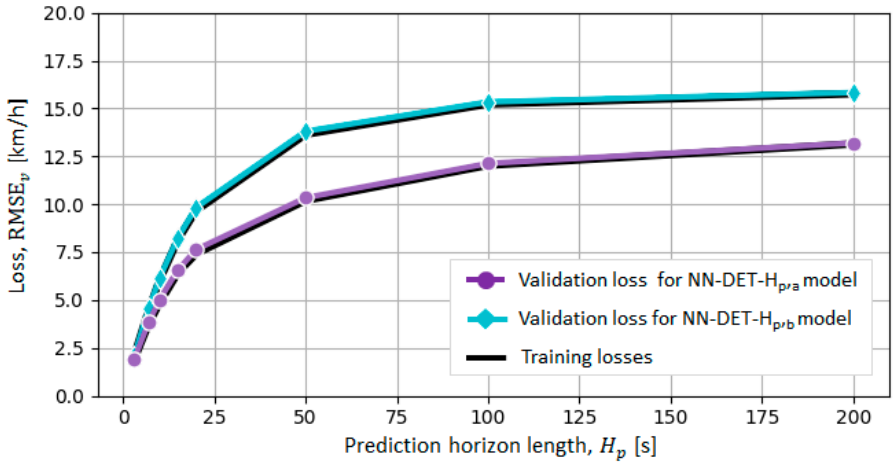

Figure 7. $RMSE_v$ loss values obtained for different prediction horizon lengths H_p of NN-DET-H$_{p,a}$ and NN-DET-H$_{p,b}$ models by using training and validation datasets.

Table 3. Training, validation, and testing results of NN-DET models with different prediction horizon length H_p [s], including the number of adjustable model parameters.

Model	Loss Value, $RMSE_v$ [km/h]			Number of NN Parameters
	Training Set	Validation Set	Test Set	
NN-DET-H$_{p,3}$	1.87 (2.05) *	1.88 (2.05)	1.90 (2.08)	49,475 (48,707)
NN-DET-H$_{p,7}$	3.74 (4.45)	3.83 (4.53)	3.84 (4.52)	49,607 (48,839)
NN-DET-H$_{p,10}$	4.84 (6.02)	5.01 (6.13)	5.00 (6.13)	49,706 (48,938)
NN-DET-H$_{p,15}$	6.42 (8.14)	6.56 (8.23)	6.59 (8.26)	49,871 (49,103)
NN-DET-H$_{p,20}$	7.42 (9.67)	7.63 (9.82)	7.58 (9.79)	50,036 (49,268)
NN-DET-H$_{p,50}$	10.20 (13.65)	10.34 (13.80)	10.35 (13.76)	51,026 (50,258)
NN-DET-H$_{p,100}$	12.04 (15.23)	12.13 (15.33)	12.15 (15.35)	52,676 (51,908)
NN-DET-H$_{p,200}$	13.15 (15.77)	13.20 (15.82)	13.22 (15.84)	55,976 (55,208)

* The loss values given in parentheses relate to single-input (NN-DET-H$_p$)$_b$ model, while the ones given outside parentheses correspond to full-input-set (NN-DET-H$_p$)$_a$ models.

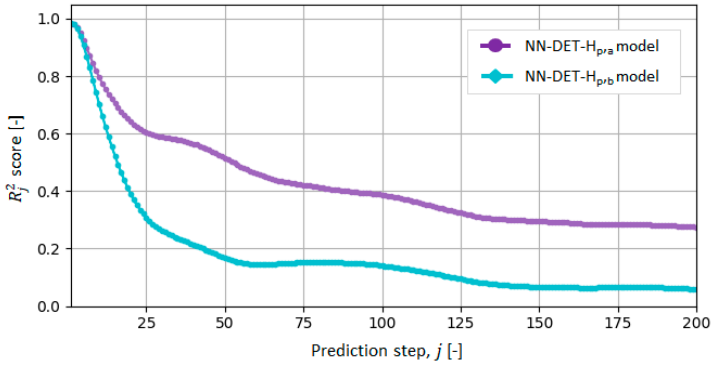

Figure 8. R_j^2 score values corresponding to jth discrete step of prediction horizon with length H_p = 200 s and calculated for validation data set (note that for given sampling time of 1 s, the prediction in-terval of, e.g., j = 200 steps correspond to the length of 200 s).

4. Stochastic Vehicle Velocity Prediction Model

To combine characteristics of the previously discussed NN-STC and NN-DET vehicle velocity prediction models, a dynamic stochastic prediction model is proposed as an ultimate solution (hereinafter referred to as NN-DSTC model). The model represents an extension of the NN-DET model with a stochastic component, which represents prediction uncertainty for each discrete time step along the prediction horizon.

4.1. Modelling of Deep Neural Network with Stochastic Output

The architecture of NN-DSTC model is also based on the deep feedforward NN, and it is shown in Figure 9. The model outputs are assumed to be the expectation $\hat{\mu}_{k,j}$ and the standard deviation $\hat{\sigma}_{k,j}$ of normal (Gaussian) velocity distributions $\mathcal{N}_{k,j}\left(\hat{\mu}_{k,j}, \hat{\sigma}_{k,j}\right)$, while the inputs to the model remain the same as in the case of NN-DET model (cf. Figure 4). In this way, the discrete velocity distributions of NN-STC model (Figure 1a) are approximated with normal distributions, thus reducing the output dimension per prediction step from $\mathbb{R}^{91 \text{ x } H_p}$ to $\mathbb{R}^{2 \text{ x } H_p}$, i.e., there are only two parameters per each prediction step $j = 1, 2, \ldots, H_p$ along the prediction horizon. The essential difference between NN-DSTC model and NN-STC model is that the former provides prediction of vehicle velocity along the prediction horizon. On the other hand, the NN-DSTC model extends upon the NN-DET model in terms of outputting the entire (normal) velocity distribution for each prediction step in addition to providing the velocity expectation.

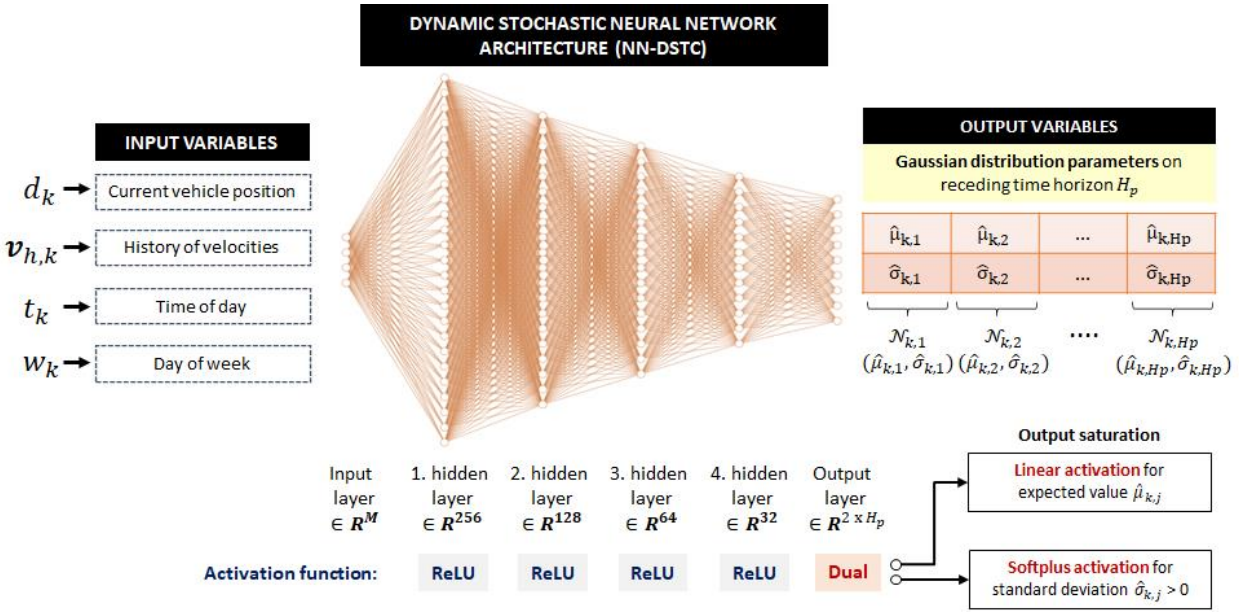

Figure 9. Proposed architecture of dynamic stochastic deep feedforward neural network (NN-DSTC) for predicting parameters of normal vehicle velocity distribution $\mathcal{N}_{k,j}\left(\hat{\mu}_{k,j}, \hat{\sigma}_{k,j}\right); j = 1, 2, \ldots, H_p$ along receding time horizon of length H_p.

The hidden layers of the NN-DSTC model again contain the *ReLU* activation function, while the output layer includes a dual activation function, i.e., *linear* one for prediction of the expectation $\hat{\mu}_{k,j}$ and *softplus* one for prediction of the standard deviation $\hat{\sigma}_{k,j}$. The latter is motivated by the need to saturate the parameter $\hat{\sigma}_{k,j}$ to values greater than or equal to zero [24]:

$$softplus\left(\hat{\sigma}_{k,j}\right) = \ln\left(1 + e^{\hat{\sigma}_{k,j}}\right) \geq 0. \tag{4}$$

One of the main challenges of training the NN-DSTC model relates to structuring the loss function, which is defined herein as the mean of negative log-likelihoods (NLL) [24] of predicted normal distributions $\mathcal{N}_{k,j}\left(\hat{\mu}_{k,j}, \hat{\sigma}_{k,j}\right)$ for each prediction step $j \in [1,\ H_p]$:

$$\begin{aligned} \overline{NLL} &= \frac{1}{N}\sum_{k=1}^{N}\frac{1}{H_p}\sum_{j=1}^{H_p} -\log\left(\mathcal{N}_{k,j}\left(\mathcal{Y}_{k,j} \,\middle|\, \hat{\mu}_{k,j}, \hat{\sigma}_{k,j}\right)\right) \\ &= \frac{1}{N}\sum_{k=1}^{N}\frac{1}{H_p}\sum_{j=1}^{H_p} -\log\left(\frac{1}{\sqrt{2\pi\hat{\sigma}_{k,j}^2}}\cdot e^{-\frac{1}{2}\left(\frac{(v_{k,j}-\hat{\mu}_{k,j})^2}{\hat{\sigma}_{k,j}^2}\right)}\right); \ \forall v_{k,j} \in \mathcal{Y}. \end{aligned} \tag{5}$$

The NN-DSTC model is again realized by using the Python libraries *Keras* and *Tensorflow*, with the same settings as in the case of the NN-DET model. For the purpose of training, validation, and testing of the NN-DSTC model, the prediction horizon length H_p is set to 10 s, while the velocity history interval length H_h is set to 20 s.

Figure 10 shows the progress of NN-DSTC model training and validation loss $\overline{NLL}$ defined by Equation (5). The validation response of $\overline{NLL}$ exhibit somewhat stronger oscillatory behaviour and spikes compared to the NN-STC model (cf. Figure 8 in [22]). This effect can be explained by the complexity of $\overline{NLL}$ loss function. Nevertheless, the absence of a notable gap between the training and validation loss curves indicates that the NN-DSTC model possesses good generalization properties. The minimum validation loss of 2.593 is reached in the 128th epoch, and the weighting coefficients obtained in that epoch are adopted as the final parameters of NN-DSTC model. The final NN-DSTC model testing loss is found to be equal to 2.598, which further confirms the good generalization properties of the model on unseen data.

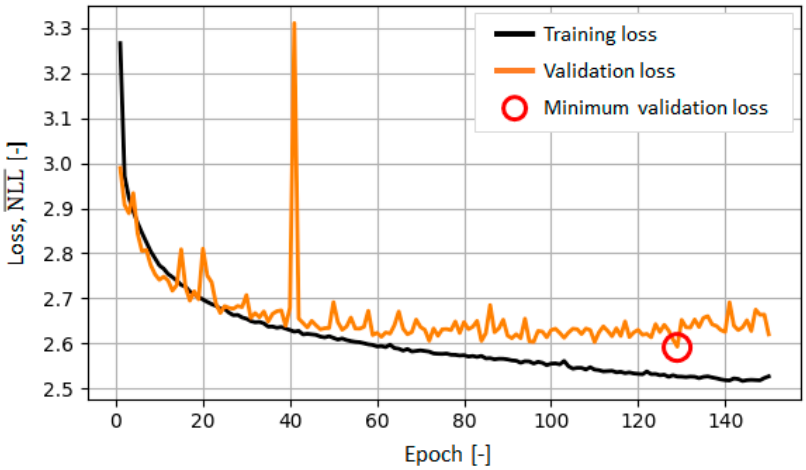

Figure 10. Progress of NN-DSTC model training and validation loss $\overline{NLL}$.

4.2. Comparative Analysis of Prediction Accuracies of Stochastic and Deterministic Models

4.2.1. Considered Metrics

A comprehensive validation of the NN-DSTC model cannot be performed unambiguously in relation to the recorded data due to the stochastic nature of the model. More specifically, validation of such a model would require knowledge of the exact values of the original vehicle velocity distribution parameters for each combination of input data, for which a virtually infinite set of recorded data would be necessary. Therefore, the validation is narrowed in this subsection to comparison of the velocity *expectation* prediction of NN-DSTC model with the prediction of NN-DET model, based on the test dataset and the following metrics:

i. Loss function $RMSE_v$ defined by Equation (2);
ii. R_j^2 score defined by Equation (3), calculated for each jth discrete step along the prediction horizon ($j = 1, 2, \ldots, H_p = 10$), as applied in Section 3;
iii. Mean value of RMSE_j, also calculated for each jth discrete step along the prediction horizon:

$$\overline{\mathrm{RMSE}_j} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left(\hat{v}_{k,j} - v_{k,j}\right)^2}; \tag{6}$$

iv. Mean prediction RMSE along the prediction horizon H_p and for the given time step k along the driving cycle:

$$\mathrm{RMSE}_k = \sqrt{\frac{1}{H_p}\sum_{j=1}^{H_p}\left(\hat{v}_{k,j} - v_{k,j}\right)^2}; \tag{7}$$

where, in the case of the NN-DSTC model, the predicted velocity expectations $\hat{\mu}_{k,j}$ are used instead of predicted velocity $\hat{v}_{k,j}$.

4.2.2. Analysis of Prediction Accuracy along the Prediction Horizon

The value of indicator $RMSE_v$, obtained for NN-DSTC model and test dataset, is equal to 5.12 km/h, which is only 2.2% higher when compared to NN-DET model of the same configuration ($RMSE_v = 5.01$ km/h; see Table 1). The certain performance degradation is illustrated in Figure 11, which shows comparative plots of the individual recorded ($v_{k,j}$) vs. predicted velocity values ($\hat{v}_{k,j}$ for NN-DET and $\hat{\mu}_{k,j}$ for NN-DSTC) for the two models. The reason for the somewhat higher prediction error of NN-DSTC model compared to NN-DET model is more complex model structure providing richer, stochastic velocity prediction (cf. Figures 4 and 9). Nevertheless, the resulting, modest increase in the prediction error of NN-DSTC model can readily be justified by the structural advantage of this model in terms of providing the extra output ($\hat{\sigma}_{k,j}$) describing the prediction uncertainty.

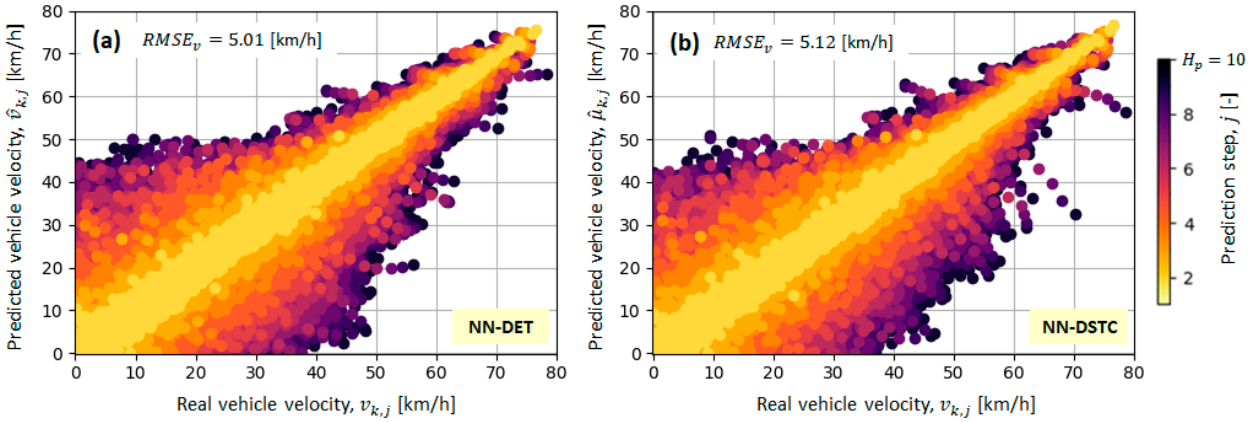

Figure 11. Predicted velocity $\hat{v}_{k,j}$ and velocity expectation $\hat{\mu}_{k,j}$ in relation to real velocity values $v_{k,j}$ for NN-DET model (**a**) and NN-DSTC model (**b**), respectively, where predictions for each jth time step along the prediction horizon H_p are marked in different colours.

Figure 11 clearly illustrates that the prediction accuracy deteriorates along the prediction horizon, i.e., the prediction error grows as the prediction step j increases towards the horizon length H_p. Additionally, the prediction error is higher at lower vehicle velocities, which is explained by more rapid vehicle accelerations in the low-speed (i.e., low-gear) range, resulting in a wider spread of velocity values to be predicted.

Figure 12 shows the R_j^2 score and $\overline{\mathrm{RMSE}}_j$ values determined at different steps $j = 1, 2, \ldots, H_p$ along the prediction horizon. These results confirm that the prediction accuracy lowers as the prediction steps j grows towards the end of prediction horizon. They also indicate that the prediction accuracy along the prediction horizon is mostly somewhat higher for NN-DET model than NN-DSTC model. The relative differences of the medians of considered metrics for the two models equals 0.24% for R_j^2 and 1.53% for $\overline{\mathrm{RMSE}}_j$, which is close to the corresponding difference of 2.2% related to $RMSE_v$ metric discussed above.

Figure 13 shows time responses of the velocity predictions provided by NN-DET and NN-DSTC models for one of the test-driving cycles. These responses confirm that the velocity prediction accuracy of NN-DSTC model is close, although generally lower than that of NN-DET model, and that the accuracy deteriorates towards the end of prediction horizon. Additionally, the prediction accuracy is higher in the case of less uncertain traffic conditions, i.e., when the predicted confidence interval $\pm\hat{\sigma}_{k,j}$ is narrower. This points to

consistent estimation of predicted velocity distribution parameter $\hat{\sigma}_{k,j}$. Finally, Figure 13 indicates that the uncertainty of vehicle velocity prediction expectedly increases as the prediction is conducted farther in future (i.e., as the prediction step j grows toward H_p). This is clearly illustrated in Figure 14 by the box plot of the predicted velocity uncertainty parameter $\hat{\sigma}_j$, which shows that the mean and median values of uncertainty parameter grow approximately linearly with the prediction step j. The same, approximately linear trend is recognized in the R_j^2 and $\overline{RMSE_j}$ plots of NN-DET model (purple curves in Figure 12). Indeed, Figure 15 illustrates that the mean $\hat{\sigma}_j$ values of NN-DTSC model (orange triangles in Figure 14) are highly correlated with R_j^2 and $\overline{RMSE_j}$ values of NN-DET model, which further confirms the accuracy of confidence interval prediction of NN-DSTC model.

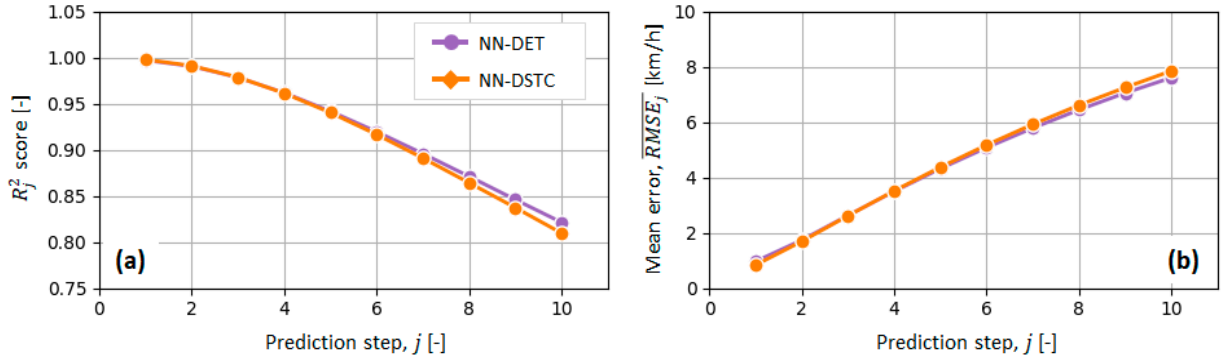

Figure 12. Comparative plots of prediction performance of NN-DET and NN-DSTC models, expressed in terms of R_j^2 (**a**) and $\overline{\mathrm{RMSE}_j}$ (**b**) metrics, determined for different prediction steps j along prediction horizon H_p.

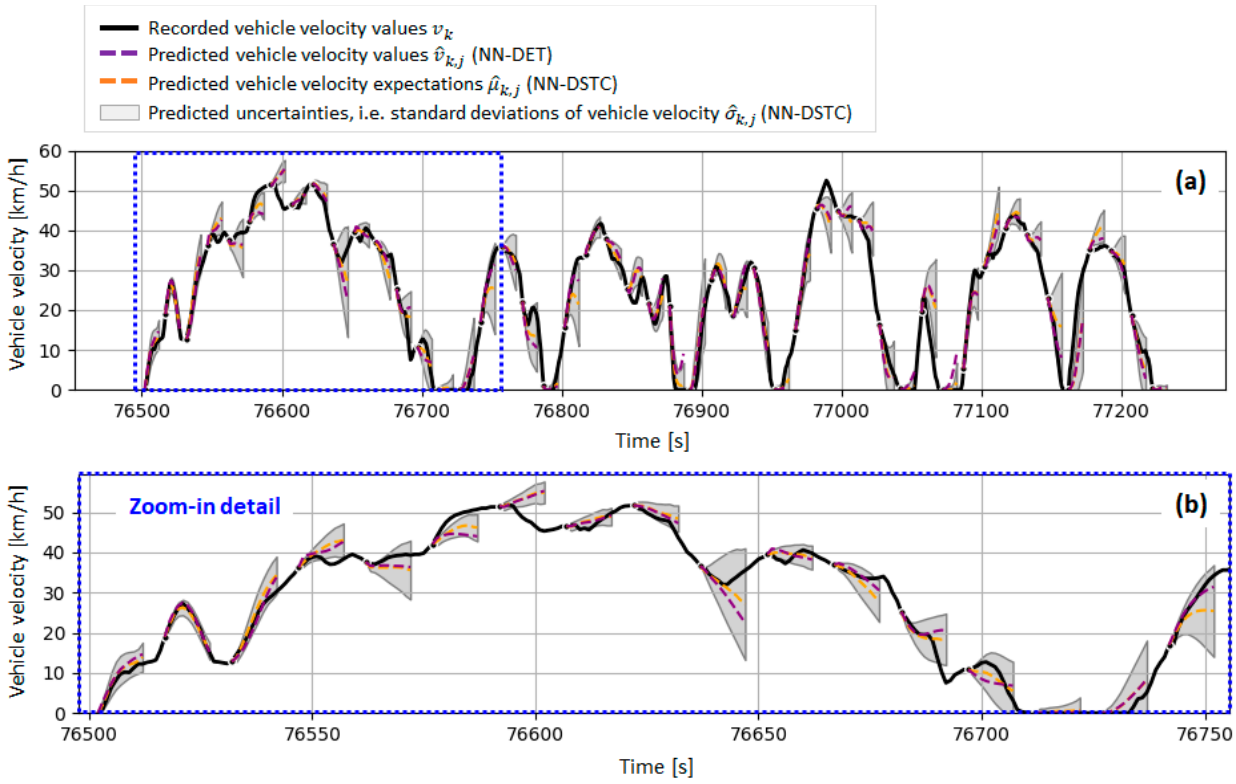

Figure 13. Comparative time response of vehicle velocity predictions provided by NN-DET and NN-DSTC models for one of driving cycles from test data set (**a**) and characteristic response zoom-in detail (**b**).

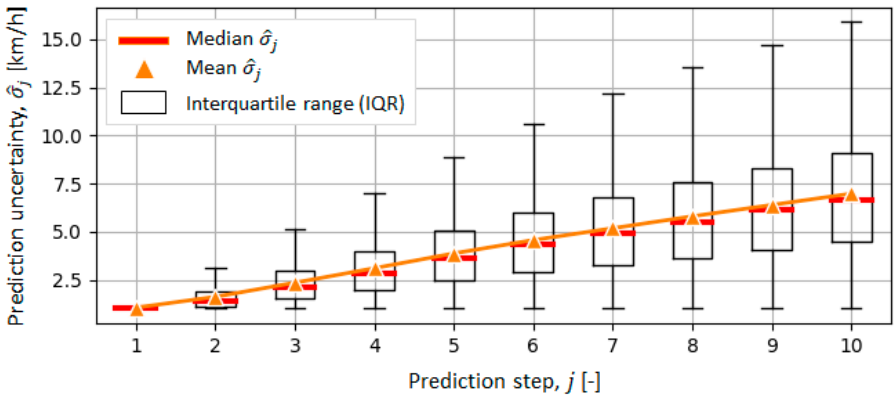

Figure 14. Boxplot of NN-DSTC model-predicted uncertainty parameter (i.e., standard deviation $\hat{\sigma}_j$) of vehicle velocity prediction along steps $j = 1, 2, \ldots, H_p$ of prediction horizon.

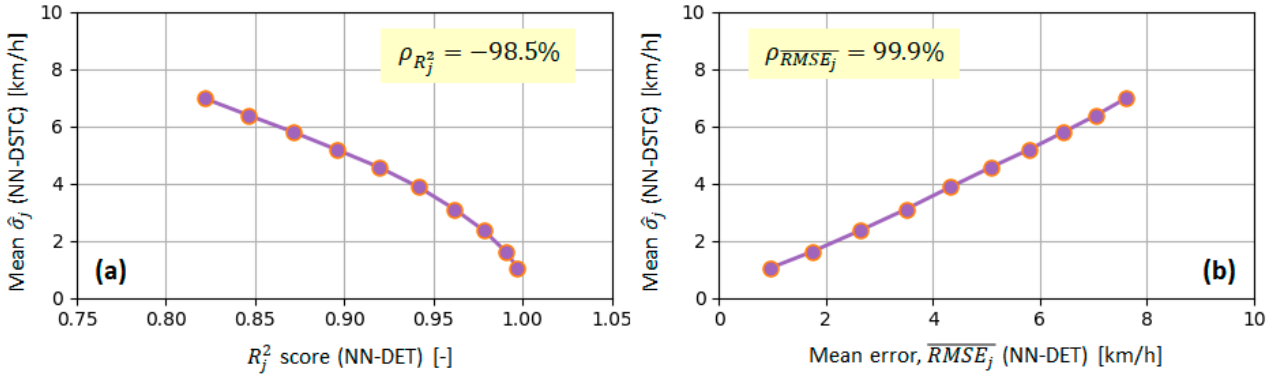

Figure 15. Dependence of NN-DET model velocity prediction accuracy metrics R_j^2 (**a**) and $\overline{RMSE_j}$ (**b**) on mean of NN-DTSC model-predicted standard deviation $\hat{\sigma}_j$, where ρ represents the corresponding Pearson correlation index value (100% means full correlation, and −100% denotes full anti-correlation).

4.2.3. Analysis of Prediction Accuracy along the Route

Figure 16a shows the individual RMS prediction errors along the route (the indicator $RMSE_k$ defined by Equation (7)) for NN-DET and NN-DSTC models (purple and orange circles, respectively). These $RMSE_k$ values are higher on road sections that have greater uncertainty of traffic conditions, i.e., on those sections for which the standard deviation σ_k of recorded test samples/population gets higher. These sections evidently correspond to the congested traffic area, characterized with low expectation μ_k of recorded test samples and zero values of confidence interval lower border $\mu_k - \sigma_k$. The two models are characterized by almost identical scattering of individual prediction errors, as well as their mean values (solid lines in Figure 16a). This again confirms the ability of more general and complex NN-DTSC model to predict velocity expectation that is close to NN-DET model velocity prediction. Figure 16b,c shows the distributions of individual RMS prediction errors for the two models. The mean value and standard deviation of $RMSE_k$ distributions are around 3.7 km/h and 3.4 km/h, respectively, for both models.

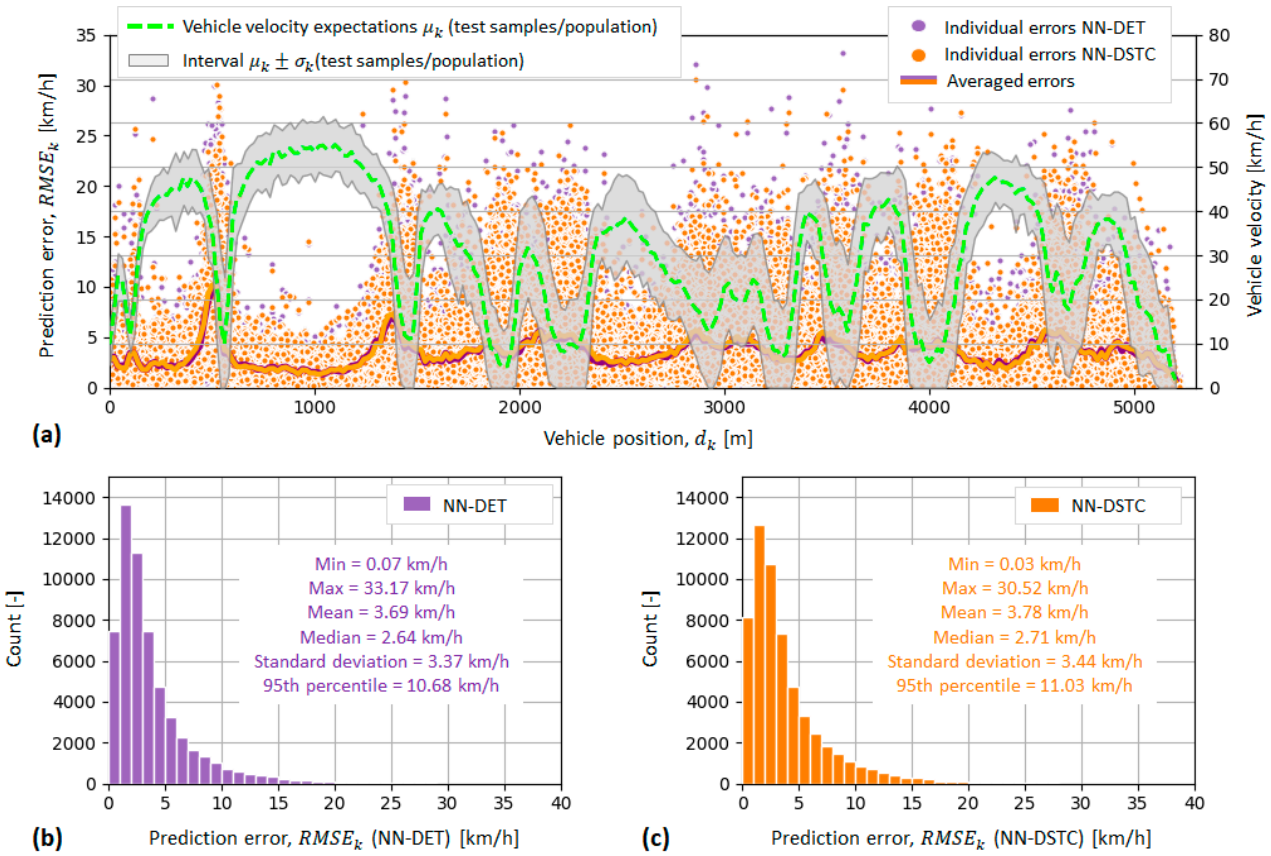

Figure 16. Scattering of NN-DET and NN-DSTC models individual RMS velocity prediction errors along the selected bus route (**a**), including the corresponding distributions given separately for NN-DET (**b**) and NN-DSTC models (**c**).

5. Conclusions

Deterministic and stochastic versions of deep neural network-based vehicle velocity prediction model have been proposed. The deterministic (NN-DET) model predicts the velocity expectation along the receding prediction horizon, while the stochastic (NN-DSTC) one additionally predicts the velocity distribution standard deviation. The proposed models have been parameterized and tested based on GPS-based tracking data recorded on several city buses during six-month period of their regular operation on a selected route.

Analysis of the influence of different model input features on the NN-DET model prediction accuracy has pointed out that the current vehicle position and short-term velocity history are the most significant input features, while the time of day and day of week information have a negligible impact (the root mean square error, $RMSE_v$, drops from 5.07 to 5.03 km/h). In addition, vehicle velocity history interval length H_h has found to have optimal value in the range from 30 to 50 s from the standpoint of high prediction accuracy on the validation/test dataset and limited model complexity. For the full-input-set NN-DET model and the prediction horizon length $H_p = 200$ s, the R_j^2 score, calculated individually for each prediction step $j = 1, 2, \ldots, H_p$ on the validation dataset, monotonically falls with j and reaches the value of approximately 0.25 for $j = H_p$, thus indicating solid accuracy throughout relatively long prediction horizons. The accuracy is significantly improved for shallow prediction sub-horizons, e.g., $R_j^2 \geq 0.8$ for $j \leq 10$ and $R_j^2 > 0.6$ for $j \leq 25$. The accuracy is considerably lower when only the velocity history input is concerned: $R_j^2 \geq 0.7$ for $j \leq 10$, $R_j^2 > 0.3$ for $j \leq 25$, and $R_j^2 \sim 0.07$ for $j = H_p = 200$. This confirms the importance of including the current vehicle position as the model input, particularly for the particular city bus application where driving cycles are related to fixed routes.

The NN-DSTC model has been found to have only slightly lower accuracy of predicting the velocity expectation along the receding horizon when compared to the NN-DET model of the same configuration (the root mean square error, $RMSE_v$, is higher by 2.2%), while offering a significant advantage of including prediction uncertainty along the horizon. The velocity expectation prediction accuracy is lower when conducted farther in future and in the case of less uncertain traffic conditions (i.e., for $RMSE_k > 3.7$ km/h), which occur typically at lower vehicle velocity expectation, i.e., for congested traffic and frequent vehicle stops. The accuracy of NN-DSTC model-predicted standard deviation of velocity normal distribution has been examined indirectly, i.e., through confirming that (i) the predicted standard deviation grows with the prediction step j, (ii) the velocity expectation accuracy is higher when the predicted standard deviation is lower, and (iii) the correlation between the mean predicted standard deviation and the NN-DET model's velocity prediction accuracy metrics along the prediction horizon expectation is higher than 98%.

The main foreseen limitation of the presented velocity prediction models and related assessment analysis is that they have been designed based on the data related to specific, although still very relevant transport system with repeating routes (city bus transport, herein). Additionally, the work is limited to feedforward NNs, while more advanced NNs may generally be more suitable for dynamic system response prediction. The future work can be directed to the following activities:

(a) Applying and examining the proposed vehicle prediction models within vehicle deterministic or stochastic model predictive control strategies (e.g., energy management strategy of a PHEV aimed at minimizing the vehicle fuel and electricity consumption for a wide range of driving cycles);
(b) Considering other types of NNs, such as recurrent NNs, which can be more suitable for the task of dynamic system behaviour prediction and potentially bring further gains in model prediction accuracy;
(c) Considering Markov chain-based stochastic velocity prediction method, which, in addition to the vehicle velocity (and acceleration), would also take information about the vehicle position when defining the Markov states;
(d) Examining the proposed prediction models for other transport systems that are not characterized by fixed/repeating routes (unlike the city bus transport system considered herein), including different types of vehicles, as well;
(e) Adding more relevant inputs to the model such as accelerator pedal opening and traffic related inputs (not available in the presented study) for potentially improved prediction performance;
(f) Comparing the various developed prediction models to each other, as well as with respect to existing models.

Author Contributions: Conceptualization, J.T., B.Š. and J.D.; methodology, J.T. and B.Š.; software, J.T.; validation, J.T., B.Š. and J.D.; writing—original draft preparation, J.T.; writing—review and editing, J.D. and B.Š.; visualization, J.T.; supervision, J.D. All authors have read and agreed to the published version of the manuscript.

Funding: It is gratefully acknowledged that this work has been done within the project ACHIEVE ("Adaptive and Predictive Control of Plug-in Hybrid Electric Vehicles"; website: http://achieve.fsb.hr/ (accessed on 15 July 2022)), supported by the Croatian Science Foundation under the Grant agreement No. IP-2018-01-8323.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are not publicly available due to privacy restrictions of related transport company.

Acknowledgments: The authors are grateful to the company Libertas d.o.o., Dubrovnik, Croatia, for providing the city bus tracking data and related technical support.

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

Abbreviations

ADAM	Adaptive Moment Estimation (optimization algorithm used for training of neural networks)
CAN	Controller Area Network
GPRS	General Packet Radio Service
GPS	Global Positioning System
MPC	Model Predictive Control
MSE	Mean Squared Error
NLL	Negative Log-Likelihood
NN	Neural Network
NN-DET	Deterministic Neural Network (model)
NN-DSTC	Dynamic Stochastic Neural Network (model)
NN-STC	Static Stochastic Neural Network (model)
PHEV	Plug-in Hybrid Electric Vehicle
RMSE	Root Mean Squared Error
TPM	Transition Probability Matrix

Table of Symbols

$\overline{MSE_v}$	Mean squared error of predicted vs. recorded vehicle velocities (used as loss function for NN-DSTC model)
$\overline{NLL}$	Mean of negative log-likelihoods of predicted normal distributions of vehicle velocity $\mathcal{N}_{k,j}\left(\hat{\mu}_{k,j}, \hat{\sigma}_{k,j}\right)$ for each prediction step $j \in [1,\ H_p]$ (used as loss function for NN-DET model)
j	Discrete step of prediction horizon, $j = 1, 2, \ldots, H_p$
k	kth discrete time instant (i.e., data sample) of given recorded driving cycle
N	Total number of data samples used to train NN-DET and NN-DSTC models
H_h	Vehicle velocity history interval length
H_p	Prediction horizon length
$\overline{RMSE_k}$	Mean prediction RMSE calculated along the full prediction horizon H_p for the kth time instant
$\overline{RMSE_j}$	Mean prediction RMSE calculated for jth discrete step of prediction horizon
R_j^2	Coefficient of determination value calculated for jth discrete step of prediction horizon
d_k	Vehicle position (i.e., distance travelled from reference route departure station) for kth time instant of given recorded driving cycle
t_k	Time of day for kth time instant of given recorded driving cycle
v_k	Vehicle velocity for kth time instant of given recorded driving cycle
w_k	Day of week for kth time instant of given recorded driving cycle
$v_{h,k}$	Vector of historical vehicle velocities of length H_h
σ_k	Vehicle velocity standard deviation calculated from test samples/population
$\hat{v}_k$	Vehicle velocity predicted by NN-DET model
$\hat{\mu}_k$	Vehicle velocity expectation predicted by NN-STC/NN-DSTC model
$\hat{\sigma}_k$	Vehicle velocity standard deviation predicted by NN-STC/NN-DSTC model
$\mathcal{X}$	Set of input (training) values
$\mathcal{Y}$	Set of output (target) values related to inputs $\mathcal{X}$

References

1. Arata, J.; Leamy, M.; Cunefare, K. Power-split HEV control strategy development with refined engine transients. *SAE Int. J. Altern. Powertrains* **2012**, *1*, 119–133. [CrossRef]
2. Jauch, C.; Tamilarasan, S.; Bovee, K.; Güvenc, L.; Rizzoni, G. Modeling for drivability and drivability improving control of HEV. *Control Eng. Pract.* **2018**, *70*, 50–62. [CrossRef]
3. Guzzella, L.; Sciarretta, A. *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2007; pp. 243–276.
4. Hrovat, D.; Cairano, S.D.; Tseng, H.E.; Kolmanovsky, I.V. The development of Model Predictive Control in automotive industry: A survey. In Proceedings of the IEEE International Conference on Control Applications, Dubrovnik, Croatia, 3–5 October 2012; pp. 295–302.

5. Rajamani, R. Longitudinal Vehicle Dynamics. In *Vehicle Dynamics and Control*, 2nd ed.; Springer: New York, NY, USA, 2006; pp. 87–111.
6. Karbowski, D.; Kim, N.; Rousseau, A. Route-based online energy management of a PHEV and sensitivity to trip prediction. In Proceeding of the 2014 IEEE Vehicle Power and Propulsion Conference (VPPC), Coimbra, Portugal, 27–30 October 2014; pp. 1–6.
7. Huang, Y.; Wang, H.; Khajepour, A.; He, H.; Ji, J. Model predictive control power management strategies for HEVs: A review. *J. Power Sources* **2017**, *341*, 91–106. [CrossRef]
8. Lemieux, J.; Ma, Y. Vehicle Speed Prediction Using Deep Learning. In Proceedings of the IEEE Vehicle Power and Propulsion Conference (VPPC), Montreal, QC, Canada, 19–22 October 2015; pp. 1–5.
9. Zhou, Y.; Ravey, A.; Marion-Pera, M.C. A survey on driving prediction techniques for predictive energy management of plug-in hybrid electric vehicles. *J. Power Sources* **2019**, *412*, 480–495. [CrossRef]
10. Sun, C.; Hu, X.; Moura, S.J.; Sun, F. Velocity Predictors for Predictive Energy Management in Hybrid Electric Vehicles. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 1197–1204.
11. Mozaffari, L.; Mozaffari, A.; Azad, N.L. Vehicle speed prediction via a sliding-window time series analysis and an evolutionary least learning machine: A case study on San Francisco urban roads. *Eng. Sci. Technol. Int. J.* **2015**, *18*, 150–162. [CrossRef]
12. Liu, K.; Asher, Z.; Gong, X.; Huang, M.; Kolmanovsky, I. *Vehicle Velocity Prediction and Energy Management Strategy Part 1: Deterministic and Stochastic Vehicle Velocity Prediction Using Machine Learning*; SAE Technical Paper, No. 2019-01-1051; SAE: Warrendale, PA, USA, 2019.
13. Li, Y.; Peng, J.; He, H.; Xie, S. The Study on Multi-scale Prediction of Future Driving Cycle Based on Markov Chain. *Energy Procedia* **2017**, *105*, 3219–3224. [CrossRef]
14. Zhang, L.; Liu, W.; Qi, B. Combined Prediction for Vehicle Speed with Fixed Route. *Chin. J. Mech. Eng.* **2020**, *33*, 60. [CrossRef]
15. Shin, J.; Sunwoo, M. Vehicle Speed Prediction Using a Markov Chain With Speed Constraints. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3201–3211. [CrossRef]
16. Gaikwad, T.D.; Asher, Z.; Liu, K.; Huang, M.; Kolmanovsky, I. *Vehicle Velocity Prediction and Energy Management Strategy Part 2: Integration of Machine Learning Vehicle Velocity Prediction with Optimal Energy Management to Improve Fuel Economy*; SAE Technical Paper, No. 2019-01-1212; SAE: Warrendale, PA, USA, 2019.
17. Rezaei, A.; Burl, J.B. Effects of Time Horizon on Model Predictive Control for Hybrid Electric Vehicles. *IFAC-Pap.* **2015**, *48*, 252–256. [CrossRef]
18. Martinez, C.M.; Hu, X.; Cao, D.; Velenis, E.; Gao, B.; Wellers, M. Energy Management in Plug-in Hybrid Electric Vehicles: Recent Progress and a Connected Vehicles Perspective. *IEEE Trans. Veh. Technol.* **2017**, *66*, 4534–4549. [CrossRef]
19. Van Duin, J.H.R.; Tavasszy, L.A.; Quak, H.J. Towards E(lectric)-Urban Freight: First Promising Steps in the Electric Vehicle Revolution. *Eur. Transp. Trasp. Eur.* **2013**, *54*, 9.
20. Sirmatel, I.I.; Geroliminis, N. Dynamical Modeling and Predictive Control of Bus Transport Systems: A Hybrid Systems Approach. *IFAC-PapersOnLine* **2017**, *50*, 7499–7504. [CrossRef]
21. Hyeon, E.; Kim, Y.; Prakash, N.; Stefanopoulou, A.G. Influence of Speed Forecasting on the Performance of Ecological Adaptive Cruise Control. In Proceedings of the ASME 2019 Dynamic Systems and Control Conference, Park City, UT, USA, 8–11 October 2019; pp. 1–8.
22. Topić, J.; Škugor, B.; Deur, J. Static Stochastic Model-Based Prediction of City Bus Velocity. In Proceedings of the 5th International Conference on Smart Systems and Technologies, Osijek, Croatia, 19–21 October 2022.
23. Topić, J.; Škugor, B.; Deur, J. Synthesis and Feature Selection-Supported Validation of Multidimensional Driving Cycles. *Sustainability* **2021**, *13*, 4704. [CrossRef]
24. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
25. Keras. Available online: https://keras.io (accessed on 18 July 2022).
26. Tensorflow. Available online: https://tensorflow.org (accessed on 18 July 2022).
27. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference for Learning Representations, San Diego, CA, USA, 7–9 May 2015.