Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 95 (2026) 433–440

Transportation Research Procedia

www.elsevier.com/locate/procedia

Euro Working Group on Transportation Annual Meeting 2025 - EWGT2025

Reconstruction of a Freeway Control Systems' Algorithm based on Convolution Neural Networks

Josephine Grau[a,*], Claude M. Weyland[a], Marvin V. Baumann[a], Peter Vortisch[a]

[a]*Karlsruhe Institute of Technology, Kaiserstraße 12, 76131 Karlsruhe, Germany*

Abstract

Freeway Control Systems (FCS) play a vital role in enhancing road safety and traffic efficiency by dynamically managing traffic through variable speed limits, overtaking restrictions, and warning messages. An accurate representation of FCS behavior in traffic simulations is essential for realistic modeling. However, the manual implementation of FCS logic in simulations is time-consuming and requires high customization. This study proposes a data-driven approach to automatically reconstruct FCS control algorithms from historical traffic and display data. The models achieved prediction accuracies of at least 87%, effectively capturing key behaviors such as congestion-related speed reductions. Among the architectures evaluated, the baseline Convolutional neural network offered the best balance of performance and computational efficiency. At the same time, more complex models showed promise for further accuracy gains with continued development. These findings demonstrate the feasibility and potential benefits of integrating FCS models based on neural networks into traffic simulations.

Keywords: freeway control systems; intelligent transport systems; traffic flow simulation; control algorithm; CNN; LSTM

1. Introduction

Freeway control systems (FCS) improve traffic safety and optimize traffic flow by influencing the driving and speed behavior of road users based on the traffic situation. FCS monitor traffic conditions and automatically display variable speed limits, overtaking restrictions, and warnings on variable message signs along the freeway. The integration of FCS into traffic flow simulations is crucial for realistic freeway modeling, enabling, for example, studies of driving behavior, autonomous driving, and further improvement of FCS control algorithms.

Although the control logic consists of straightforward rule-based algorithms, a manual reimplementation faces several fundamental challenges. First, it is highly dependent on detailed knowledge of the underlying control algorithm. These algorithms are rarely fully documented, as they have often evolved historically through local adaptations and

* Corresponding author. Tel.: +49 721 608 42254
E-mail address: josephine.grau@kit.edu

10.1016/j.trpro.2026.02.055

parameter changes. The opaque implementation details make it difficult and time-consuming to accurately replicate the control algorithm. Secondly, manual reimplementation is tailored to the specific freeway section, with its local FCS algorithm and system design, which limits its applicability to other contexts and requires repeated development efforts for new scenarios. This implementation effort is often prohibitive for users whose research does not directly target FCS, but who require it in their models simply to ensure realistic surrounding traffic flow, resulting in their neglect or oversimplification in modeling.

This study addresses these challenges by employing neural networks to automatically reconstruct the control algorithm from historical traffic and display data. These networks learn the relationships between traffic conditions and the displayed messages. Since neural networks learn from input-output data, they can replicate the system behavior even if the underlying rules are unknown. This approach can be applied to other freeway sections using new data.

From a theoretical perspective, this approach examines how effectively neural networks can approximate the complex spatiotemporal dependencies of traffic data and control decisions, offering insights into the representational capabilities of machine learning in traffic system modeling. This study not only demonstrates the feasibility of neural network-based FCS reconstruction but also raises the broader research question of whether data-driven models can generalize and robustly capture the operational strategies of complex cyber-physical systems.

1.1. Freeway Control Systems

Freeway control systems are widely installed on freeways with high traffic volumes to actively manage traffic flow in real-time. These systems consist of measurement sections that collect traffic data and display sections that communicate control measures, such as speed limits and warnings, on LED panels spaced every 1 to 2 kilometers. Their goals are to enhance safety and maintain a steady traffic flow. The warnings about congestion, construction sites, accidents, or weather conditions, combined with dynamic speed limits, have been shown to reduce both the frequency and severity of accidents and increase roadway capacity (Riegelhuth and Glatz (2015)).

The typical operation of an FCS involves the continuous collection of traffic and environmental data, which is evaluated at fixed intervals, e.g., every minute, by a control algorithm that determines the appropriate messages to display. In Germany, these control algorithms are governed by the MARZ guidelines (Federal Ministry of Transport and Infrastructure (Ed.) (2018)), which define control logic based on threshold values for various traffic indicators such as speed, volume, and truck share. This input data is separately collected for passenger cars (PC) and heavy-duty vehicles. The MARZ framework specifies modular, parametrizable algorithms that can be adapted to local conditions. Common control modules include dynamic speed limits, truck overtaking bans, congestion warnings, and other hazard-related messages. Core control principles include longitudinal coordination to ensure consistent messaging across sections, prioritization among competing control modules, and intentional switching delays to avoid excessive message changes.

Beyond MARZ, other control strategies have emerged in practice and research. This study employs the SARAH algorithm, an extension of MARZ, which utilizes lane-level rather than cross-section-level traffic data. The optimization of dynamic speed limits has been a subject of research for several years. An overview of the state-of-the-art is given in Zhang et al. (2024). For example, Hegyi et al. (2008) developed the SPECIALIST algorithm, which reduces upstream-propagating congestion waves based on shockwave theory. More recently, reinforcement learning has been applied to optimize control strategies, as reviewed in Kušić et al. (2020) and Zhang et al. (2024).

Despite the significance of FCS, they are rarely integrated into microscopic traffic simulations, except for relatively short sections when testing new control algorithms. One exception is the detailed manual reconstruction of the A5 freeway's FCS by Weyland (2023), which served as a virtual testbed in this study. Using synthetic data generated from that simulation, our model learns to replicate the control logic based solely on traffic and display data. Since speed limits most directly impact traffic flow and can be represented in simulations, the model focuses on replicating these, while excluding warnings (with the exception of congestion warnings, which imply speed limits) and overtaking bans.

1.2. Models to capture traffic dynamics

Reconstructing control algorithms from traffic and display data requires models that can accurately capture complex traffic dynamics. This has already been investigated for the purpose of traffic prediction and congestion detection.

Recent works include Metzger et al. (2025) using a bidirectional Long-Short-Term Memory (LSTM) network to classify congestion patterns on freeways. Guo et al. (2019) demonstrated that Graph Convolutional Network (GCN) architectures can enhance traffic flow forecasting. Yu et al. (2018) further extended this approach with Spatio-Temporal Graph Convolutional Networks (ST-GCNs), capturing both spatial and temporal dependencies in traffic networks. Similarly, Li et al. (2018) introduced Diffusion Convolutional Recurrent Neural Networks (DCRNNs), which extend the convolution operation from regular grids to graphs and keep space-time relations.

The architecture proposed in this study builds on these ideas, combining convolutional and recurrent components to effectively learn the temporal and spatial patterns necessary to reconstruct freeway control behavior from data.

2. Methods

2.1. Data

In contrast to real-world traffic data, simulated traffic data offers the advantage of being complete, consistent, and free from measurement errors or irregularities caused by special events such as accidents or construction work. This makes it particularly suitable for the initial development and validation of a reconstruction model.

For this study, we used a calibrated microscopic traffic flow simulation with a manually implemented FCS based on the SARAH control algorithm. The simulation models a 30-kilometer section of the A5 freeway near Frankfurt am Main, Germany (Weyland (2023)). The simulation covered two distinct time windows on September 18, 2019: a morning peak period characterized by congestion (5–11 a.m.) and a less congested afternoon period (11 a.m.–3 p.m.). Both driving directions were considered, resulting in four representative traffic scenarios.

Each scenario was simulated ten times using the PTV Vissim microscopic traffic simulator. The resulting data served as labeled input for training the neural networks. Nine out of ten simulation runs per scenario were used for model development, with a two-thirds/one-third split for training and testing, respectively. The remaining run was reserved for validation.

2.2. Models

A key challenge in training these neural networks is accounting for control mechanisms that are not directly data-driven, such as switching delays and the longitudinal coordination of display messages. To address this, models must accurately capture spatio-temporal relationships in the traffic and control data. Convolutional Neural Networks (CNNs), commonly used in computer vision tasks, are well suited for detecting spatial dependencies, interpreted here as patterns across neighboring time steps and locations (see Figure 2).

In this study, we designed a CNN architecture (Goodfellow et al. (2016)) that begins with an initial convolutional layer using 64 filters to capture a wide range of features from the input data. This is followed by 20 convolutional layers, each with 32 filters and ReLU activation functions. A global max pooling layer and a dense output layer (Chollet and contributors (2015)) were appended at the end of the network. This model serves as the baseline and will be referred to as the "pure CNN" or simply "CNN" in the following sections.

To enhance the model's ability to learn temporal dependencies, we extended this architecture by incorporating a Long Short-Term Memory (LSTM) layer (Hochreiter and Schmidhuber (1997)). LSTMs, a type of recurrent neural network, are designed to model sequences and were applied here along both the time and spatial dimensions. This was achieved using the TimeDistributed wrapper, which allows the LSTM to process each time step — or, after transposing the axes, each spatial position — independently (Chollet and contributors (2015)). The complete architecture of the CNN-LSTM model is detailed in Table 1.

The problem was framed as a multi-class classification task with the output classes defined as: no speed restriction, 120, 100, 80, 60, and congestion warning. Accordingly, the categorical cross-entropy loss function was used. The Adam optimizer (Kingma and Ba (2017)) was chosen for its efficiency and adaptive learning rate properties. To reduce computation time and prevent overfitting, an early stopping mechanism was implemented, terminating training when no improvement in validation loss was observed over a number of three epochs.

The input features comprise minute-wise traffic measurements, including average speed and volume data, separately recorded for passenger cars and heavy-duty vehicles, as well as detector occupancy rates. Data from the leftmost

Table 1: Overview of the CNN-LSTM model architecture

Layer #	Layer Type	Details	Activation
1	Input	—	—
2	Conv2D	64 filters, (2×2) kernel	ReLU
3–22	Conv2D (×20)	32 filters, (2×2) kernel	ReLU
23	TimeDistributed LSTM	32 units, return sequences	ReLU
24	Permute	Axes (2, 1, 3)	—
25	TimeDistributed LSTM	32 units, return sequences	ReLU
26	Permute	Axes (2, 1, 3)	—
27	GlobalMaxPooling2D	—	—
28	Dense	6 units	Softmax

lane and the aggregated values across the cross-section were both included. The cross-section number was added as an additional input feature to allow the model to learn location-specific control behavior. Through iterative experimentation, the optimal input window was found to include data from five consecutive cross-sections and five minutes of temporal history. The task of the model is to predict the FCS state for the following minute at the third cross-section — that is, using historical data from two upstream, two downstream, and the target (central) cross-section, as shown in Figure 1.

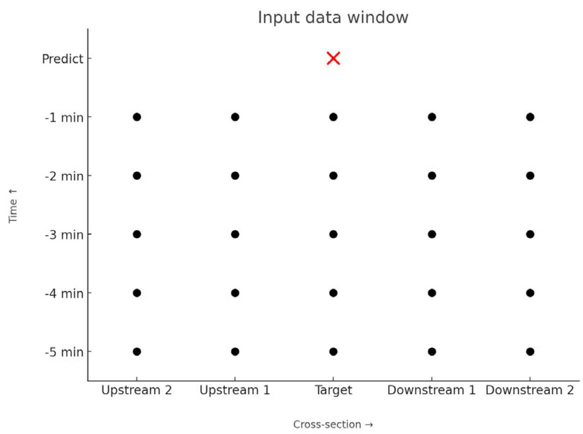


Fig. 1: Schematic representation of the model's input window

To further support learning of the switching delay mechanism, a modified version of the CNN model was developed by including the FCS status from the previous time step as an additional input. Training of this model began with teacher forcing, in which the model was provided with the ground truth FCS status from the prior time step. Gradually, scheduled sampling (Bengio et al. (2015)) was employed to transition the model from relying on ground truth inputs to using its own previous predictions. In the final stage of training — the last fifth of epochs — the model relied entirely on its own outputs, thus emulating real inference conditions.

2.3. Evaluation methods

Multiple evaluation methods were employed to assess the performance of the proposed models, encompassing general classification metrics, confusion matrix analysis, and traffic-related evaluation techniques. The following section provides a detailed explanation of these methods.

A confusion matrix provides a detailed view of classification performance by visualizing how predicted classes align with the true classes. It enables in-depth analysis of misclassifications and helps to identify systematic patterns in prediction errors. Confusion matrices can be evaluated for the whole regarded freeway section or a single cross-section only. Figure 4 presents the confusion matrices for the models tested. The values are normalized with respect to the true classes, meaning that each entry in the matrix represents the percentage of predictions assigned to class x given the true class y.

To quantify the predictive performance, three key statistical measures were employed: accuracy, Matthews' correlation coefficient (MCC), and Cohen's kappa. These metrics are widely used in multi-class classification problems. Additionally, the mean squared error (MSE) was computed to account for the ordinal nature of the output classes, assigning lower penalties to near-miss predictions (e.g., confusing 60 with 80) and higher penalties to more distant misclassifications (e.g., 60 with 120).

- **Accuracy:** Accuracy is defined as the ratio of correctly classified instances to the total number of instances. Although straightforward, it may be misleading in the presence of a class imbalance. For instance, in the morning peak hour in the southbound direction, only 4.8% of the samples displayed a speed limit of 60, while 33.6% displayed 120, potentially skewing the accuracy measure.
- **Matthews Correlation Coefficient (MCC):** MCC provides a more balanced measure of classification performance, particularly suitable for imbalanced datasets. In the multi-class case, MCC ranges from a minimum value (between −1 and 0, depending on the class distribution) to +1, indicating perfect prediction. In binary classification, a score of −1 represents complete misclassification. For multi-class classification, the MCC is computed using the $K \times K$ confusion matrix C as follows Gorodkin (2004):

$$MCC = \frac{cs - \vec{t} \cdot \vec{p}}{\sqrt{s^2 - \vec{p} \cdot \vec{p}}\sqrt{s^2 - \vec{t} \cdot \vec{t}}}$$

where:

$$t_k = \sum_i C_{ik} \quad \text{(number of times class } k \text{ truly occurred)}$$

$$p_k = \sum_i C_{ki} \quad \text{(number of times class } k \text{ was predicted)}$$

$$c = \sum_k C_{kk} \quad \text{(number of correctly predicted instances)}$$

$$s = \sum_i \sum_j C_{ij} \quad \text{(total number of samples)}$$

- **Cohen's Kappa** (Cohen (1960))**:** Cohen's kappa quantifies the agreement between predicted and actual classes while accounting for chance agreement. A kappa value of 1 indicates perfect agreement, while a value near 0 suggests that the level of agreement is no better than random chance.

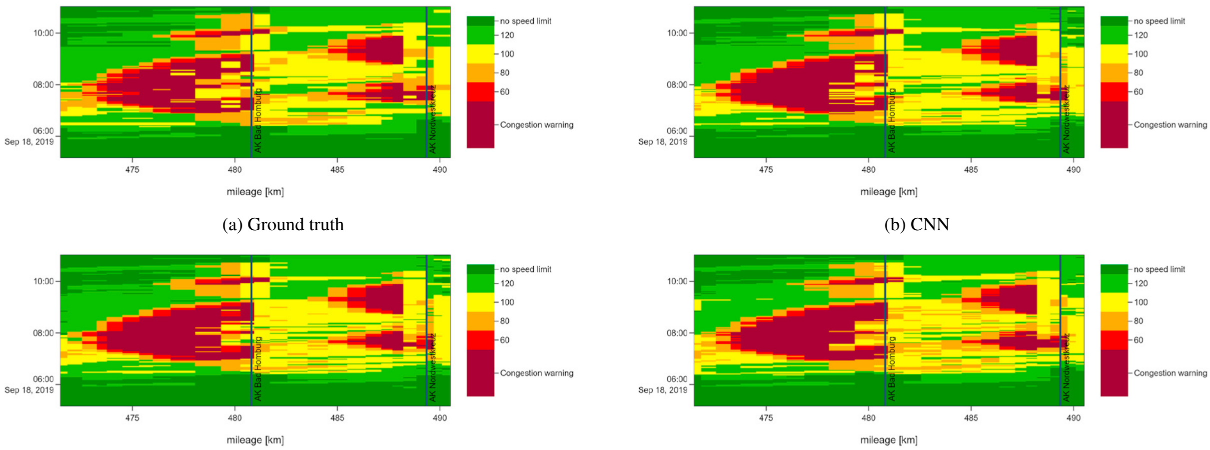
In addition to standard classification metrics, traffic-specific evaluation techniques were applied to assess the system's effectiveness in the domain of traffic monitoring and modeling.

Fundamental diagrams, plotting passenger car (PC) traffic volume on the x-axis and PC velocity on the y-axis, were used to analyze the relationship between traffic state and the actual or predicted control state. These diagrams provide insight into how well the reconstruction model recognizes traffic states. Figure 3 presents a fundamental diagram for a representative cross-section (cross-section 54) within the study area, illustrating the expected strong correlation between traffic conditions and the displayed speed limits.

Space-time diagrams were utilized to visualize the spatiotemporal evolution of traffic flow. By plotting vehicle trajectories over time and space, these diagrams facilitate an understanding of congestion formation, propagation, and dissipation within the monitored area. Here, instead of vehicle trajectories, the states of the FCS are plotted (see

Figure 2). These diagrams originally inspired the use of convolutional neural networks as they show the space-time relations in a 2D picture.

3. Results



(a) Ground truth (b) CNN

(c) CNN-LSTM (d) Sequential CNN

Fig. 2: Comparison of the Space-time diagrams of FCS states

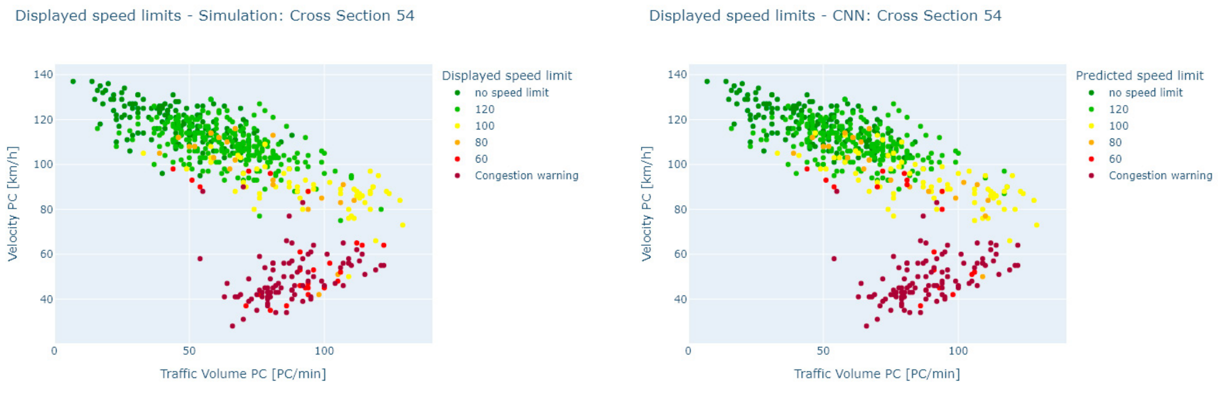
To gain an initial understanding of the model performance, the space-time diagrams provide a clear and informative visualization. Figure 2 presents the ground truth FCS states alongside the outputs of the reconstructed algorithms from the three evaluated models. Overall, the general traffic conditions are well captured, with all models successfully identifying congestion patterns. This is further supported by the fundamental diagrams at cross-section 54 (located near kilometer 487), as shown in Figure 3. All three models generally reduce speed to 60 or 80 km/h upstream of a congestion warning, consistent with the behavior embedded in the original control algorithm. Figure 3 also illustrates that the 80 km/h limit is frequently applied at moderate traffic volumes (e.g., around 70 PC/min at 110 km/h), indicating that this limit is often used not in direct response to congestion but as a preventive or safety-related measure. This difference between congestion-driven and precautionary speed limits, which was mostly correctly captured by the models, highlights their ability to recognize both traffic-dependent and space-time dependent control actions.

Despite overall good performance, correctly predicting the 80 km/h and 60 km/h speed limits remains challenging for all models. Speed reduction upstream of a congestion warning is not enforced as a logic condition, and exceptions occur — for instance, the CNN does not show a reduction to 60 km/h at kilometer 477 around 8:50 a.m. The CNN-LSTM model shows slight improvements in predicting the 60 and 80 km/h speed limits, with higher accuracy for these classes as evident in the confusion matrices (see Figure 4). The Sequential CNN model predicts the 80 km/h limit more frequently than the others. This behavior suggests that the model is more sensitive to switching delays, a control logic characteristic that the other models may not as effectively capture. As a result, the Sequential CNN exhibits switching patterns that more closely resemble the ground truth, with fewer erratic transitions, especially around kilometer 484.

Both the pure CNN and CNN-LSTM models tend to misclassify 120 km/h as 100 km/h more frequently, as seen in the confusion matrix (Figure 4) and the fundamental diagram (Figure 3). The Sequential CNN reduces this specific error, but does so at the cost of overpredicting 80 km/h speed limits.

To provide a quantitative comparison of model performance, Table 2 summarizes the evaluation metrics for all tested architectures. Each model was trained for up to 50 epochs, with early stopping applied when no further improvement in validation loss was observed. All experiments were conducted on a system equipped with an AMD EPYC 7343 16-core processor, running at 3.2 GHz, which offers 16 physical cores and 32 logical threads.

The baseline CNN model achieved strong overall performance, with accuracy values around 88% and a Matthews Correlation Coefficient (MCC) exceeding 0.80. The CNN-LSTM model yielded marginal improvements over the pure



(a) Ground Truth (b) CNN

Fig. 3: Comparison of Ground Truth snd CNN Fundamental diagrams

(a) CNN (b) CNN-LSTM (c) Sequential CNN

Fig. 4: Comparison of the Confusion matrices of the three models

CNN in terms of accuracy and agreement metrics such as MCC and Cohen's kappa. However, this came at the cost of a slightly higher Mean Squared Error (MSE) (84.60 vs. 80.23), suggesting a trade-off in how precisely the model differentiates between closely related output classes.

The Sequential CNN model underperformed across all metrics compared to the CNN and CNN-LSTM models. Nevertheless, after 50 epochs, it reached an accuracy of 87.20% and an MCC of 0.79, narrowing the gap in classification performance. However, this improvement came at additional computational costs: training times exceeded five hours, compared to just several minutes for the CNN and CNN-LSTM models. For the Sequential CNN to become a viable alternative, improvements in architecture and training efficiency would be essential.

Table 2: Model performance metrics.

Model name	**accuracy**	**MCC**	**Cohen's kappa**	**MSE**	**epochs**	**training time [s]**
CNN	0.8764	0.7936	0.7925	86.2906	10	223
CNN	0.8809	0.8019	0.8016	80.2334	19 (early stopping)	412
CNN-LSTM	0.8796	0.8000	0.7997	83.5454	10	386
CNN-LSTM	0.8825	0.8048	0.8047	84.6013	16 (early stopping)	612
Sequential CNN	0.8664	0.7770	0.7755	108.3745	10	3539
Sequential CNN	0.8720	0.7878	0.7876	90.8363	50	18304

4. Conclusion and Future Research

All models demonstrated the ability to reconstruct key aspects of freeway control behavior, e.g. congestion-related speed reductions. The accuracy of FCS state prediction reaches at least 87% across all evaluated models. Employment of these models in traffic flow simulations would likely improve simulation significance compared to the usage of static speed limits. While the CNN-LSTM offered slight accuracy gains, and the Sequential CNN captured control logic more realistically, the baseline CNN remains the most balanced choice in terms of performance and efficiency. Further refinement of architecture and training of the CNN-LSTM and the Sequential CNN models might fully exploit their potential.

Further improvements and research directions are envisioned to enhance the system's capabilities and extend its applicability. The following areas are identified as key next steps:

- **Applicability and Robustness**: A key next step is to assess the impact of minor deviations (e.g., 10% variations in displayed speed limits) on traffic dynamics in simulation. This will help evaluate whether the current level of predictive accuracy is sufficient for practical applications. Furthermore, validating the model on real-world datasets is essential to test generalizability under realistic, noisy conditions.
- **Model Architecture:** Future work should explore alternative model architectures to determine whether they can further improve performance. In addition, it will be important to assess whether models tailored to the SARAH algorithm used on the A5 freeway can generalize to other FCS control algorithms.
- **Improving Explainability:** As neural networks often operate as black boxes, enhancing model interpretability is crucial — particularly in safety-critical applications such as freeway control. Future research will focus on integrating techniques from the field of Explainable AI (e.g., Montavon et al. (2018)).

References

Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N., 2015. Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks.

Chollet, F., contributors, 2015. Keras layers. https://keras.io/api/layers.

Cohen, J., 1960. A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20, 37–46.

Federal Ministry of Transport and Infrastructure (Ed.), 2018. Merkblatt für die Ausstattung von Verkehrsrechnerzentralen und Unterzentralen (Guidelines for the Equipment of Traffic Control Centers and Subcenters).

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press. Http://www.deeplearningbook.org.

Gorodkin, J., 2004. Comparing two K-category assignments by a K-category correlation coefficient. Computational Biology and Chemistry 28, 367–374.

Guo, S., Lin, Y., Feng, N., Song, C., Wan, H., 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. Proceedings of the AAAI Conference on Artificial Intelligence 33, 922–929.

Hegyi, A., Hoogendoorn, S., Schreuder, M., Stoelhorst, H., Viti, F., 2008. SPECIALIST: A dynamic speed limit control algorithm based on shock wave theory, in: 2008 11th International IEEE Conference on Intelligent Transportation Systems, IEEE, Beijing, China. pp. 827–832.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Computation 9, 1735–1780.

Kingma, D.P., Ba, J.L., 2017. Adam: A method for stochastic optimization, in: ICLR 2015.

Kušić, K., Ivanjko, E., Gregurić, M., Miletić, M., 2020. An Overview of Reinforcement Learning Methods for Variable Speed Limit Control. Applied Sciences 10, 4917.

Li, Y., Yu, R., Shahabi, C., Liu, Y., 2018. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting.

Metzger, B., Kessler, L., Kolb, M., Bogenberger, K., 2025. Spatio-temporal prediction of freeway congestion patterns using neural networks.

Montavon, G., Samek, W., Müller, K.R., 2018. Methods for interpreting and understanding deep neural networks. Digital Signal Processing 73, 1–15.

Riegelhuth, G., Glatz, M., 2015. Zuverlässiger Betrieb von Streckenbeeinflussungsanlagen auf Basis einer antizipierenden, regelbasierten Steuerung (Reliable operation of traffic control systems based on anticipatory, rule-based control). Straßenverkehrstechnik , 245–258.

Weyland, C.M.L., 2023. Mikroskopische Verkehrsflusssimulation von Autobahnen mit Streckenbeeinflussungsanlagen (Microscopic Traffic Flow Simulation with Freeway Control Systems). Ph.D. thesis. Karlsruhe Institute of Technology.

Yu, B., Yin, H., Zhu, Z., 2018. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, pp. 3634–3640.

Zhang, Y., Quinones-Grueiro, M., Zhang, Z., Wang, Y., Barbour, W., Biswas, G., Work, D., 2024. Marvel: Bringing multi-agent reinforcement-learning based variable speed limit controllers closer to deployment. IEEE Access 12, 161995–162014.