

Transportation Research Record 1–12 © National Academy of Sciences: Transportation Research Board 2020 Article reuse guidelines: sagepub.com/journals-permissions DOI: 10.1177/0361198120911052 journals.sagepub.com/home/trr



Two-Stream Multi-Channel Convolutional Neural Network for Multi-Lane Traffic Speed Prediction Considering Traffic Volume Impact

Ruimin Ke¹, Wan Li², Zhiyong Cui¹, and Yinhai Wang^{1,3}

Abstract

Traffic speed prediction is a critically important component of intelligent transportation systems. Recently, with the rapid development of deep learning and transportation data science, a growing body of new traffic speed prediction models have been designed that achieved high accuracy and large-scale prediction. However, existing studies have two major limitations. First, they predict aggregated traffic speed rather than lane-level traffic speed; second, most studies ignore the impact of other traffic flow parameters in speed prediction. To address these issues, the authors propose a two-stream multi-channel convolutional neural network (TM-CNN) model for multi-lane traffic speed prediction considering traffic volume impact. In this model, the authors first introduce a new data conversion method that converts raw traffic speed data and volume data into spatial–temporal multi-channel matrices. Then the authors carefully design a two-stream deep neural network to effectively learn the features and correlations between individual lanes, in the spatial–temporal dimensions, and between speed and volume. Accordingly, a new loss function that considers the volume impact in speed prediction is developed. A case study using 1-year data validates the TM-CNN model and demonstrates its superiority. This paper contributes to two research areas: (1) traffic speed prediction, and (2) multi-lane traffic flow study.

Traffic speed prediction is one of the most crucial components of intelligent transportation systems. It can benefit both traffic agencies and travelers by contributing to key applications such as variable speed limit control and route guidance. Although traffic speed prediction has a long history that can be dated back several decades, traditional traffic speed prediction methods are unable to precisely capture the high dimensional and nonlinear characteristics of traffic flow because of the lack of either computational ability or amount of data (1, 2). In recent years, with the emerging trends in artificial intelligence and transportation data science, a growing body of research has been conducted in this field.

The typical traffic speed prediction problem is to predict traffic speed at a future time using given historical traffic data. Traditionally, time series methods such as autoregressive integrated moving average (ARIMA) and conventional machine learning models such as support vector regression (SVR) are widely applied to traffic prediction (3–12). Later on, with the tremendous success of deep learning in many fields (13–15), researchers started to explore the possibility of deep learning for traffic speed prediction and then developed a couple of deep learning models that significantly outperformed the conventional models (16–21). For example, Ma et al. implemented a long short-term memory neural network (LSTM NN) for the first time in traffic speed prediction. Their work suggested that LSTM NN produced the best performance over previous methods (19). Tang et al. designed an improved fuzzy neural network for traffic speed prediction. This model considered the periodic characteristics of traffic flow and achieved state-of-the-art performance (20). Although these pioneering studies still focus on relatively small-scale prediction at individual locations, they have greatly inspired the explorations of more advanced deep-learning-based traffic speed prediction methods.

Recently, substantial research has focused on extending the traffic speed prediction problem from individual

³Pacific Northwest Transportation Consortium, U.S. DOT University Transportation Center for Federal Region 10, Seattle, WA

Corresponding Author:

Yinhai Wang, yinhai@uw.edu

¹Smart Transportation Application and Research Lab, Department of Civil and Environmental Engineering, University of Washington, Seattle, WA ²Oak Ridge National Laboratory, Knoxville, TN

roadway locations to traffic networks by designing new deep neural networks that integrate physical roadway structures (22-30). Ma et al. developed a convolutional neural network (CNN) model that can capture spatial correlations between adjacent roadway segments and temporal correlations between adjacent times in a 2D spatial-temporal matrix (22). Yao et al. proposed a deep learning architecture named spatial-temporal dynamic network (STDN) that incorporated CNN, LSTM, and a periodically shifted attention mechanism to address the issues on dynamic dependency and shifting of long-term periodic dependency (29). Cui et al. devised a high-order graph convolutional LSTM (HGC-LSTM) to model the dynamics of the traffic speed and acquire the spatial dependencies within the traffic network. This group of studies considers both spatial dependencies and temporal dynamics of traffic flow in deep learning models, and thereby enables effective learning and accurate speed prediction for network-scale traffic.

Despite the achievements mentioned above in traffic speed prediction, the existing studies have two major limitations. First, they predict aggregated traffic speed rather than lane-level traffic speed. At every data collection unit of roadways, they implicitly assume no traffic pattern difference between different lanes. In some studies, this is because of the unavailability of lane-level traffic data; in others in which the lane-level data are available, the speeds are still often aggregated to simplify the model complexity. However, since over three decades ago, research has revealed that traffic flows on different lanes show different vet correlated patterns (31–39). For instance, Daganzo's work studied a "reverse lambda" pattern (37). This pattern shows as consistently high flows on freeway median lanes, but it has not been reported for the shoulder lanes. It is also observed that for either two-lane or three-lane freeway segments, there are certain volume-density distributions for individual lanes (34). As there is an increasing need for lane-based traffic operations such as carpool lane tolling and reversible lane control in modern transportation systems, this issue can no longer be ignored.

The second limitation is that most existing studies ignore other traffic flow parameters in speed prediction tasks. In traffic flow theory, there are correlations among traffic flow speed, volume, and occupancy (40). Without the integration of volume or occupancy into speed prediction, the hidden traffic flow patterns may not be fully captured and learned, which can lead to reduced prediction accuracy (41). An intuitive example is that: In freeflow conditions, a larger-volume traffic stream tends to be more sensitive to perturbances than a smaller-volume traffic stream. Therefore, the speed of the larger-volume traffic stream is more likely to decrease in the future time. However, without the volume or occupancy data, it is hard to model the hidden traffic flow patterns.

To address these challenges, the authors propose a two-stream multi-channel convolutional neural network (TM-CNN) for multi-lane traffic speed prediction with the consideration of traffic volume impact. In the proposed model, the authors develop a data conversion method to convert both the multi-lane speed data and multi-lane volume data into multi-channel spatialtemporal matrices. A CNN architecture with two streams is designed, in which one takes the multi-channel speed matrix as input and another takes the multi-channel volume matrix as input. A fusion method is further implemented for the two streams. Specifically, convolutional layers learn the two matrices to capture traffic flow features in three dimensions: the spatial dimension, the temporal dimension, and the lane dimension. Then, the output tensors of the two streams will be flattened and concatenated into one speed-volume vector, and this vector will be learned by the fully connected (FC) layers. Accordingly, a new loss function is devised considering the volume impact in the speed prediction task.

The proposed TM-CNN model is validated using 1-year loop detector data on a major freeway in the Seattle area. The comprehensive comparisons and analyses demonstrate the strength and effectiveness of the model. This paper contributes to two transportation research areas. First, it contributes to the traffic speed prediction area by adding a new deep neural network model to the existing literature. Second, it pushes off the boundary of knowledge in the multi-lane traffic flow study area by developing a method for the learning and speed prediction of multi-lane traffic. In summary, the contribution of this paper is fourfold:

- (1) The authors introduce a new data conversion method to convert the multi-lane traffic speed data and volume data into spatial-temporal multi-channel matrices. The converted data matrices are organized as the inputs to the deep neural network.
- (2) A two-stream CNN architecture for multi-lane traffic speed prediction is designed. The convolutional layers extract the correlations between lanes and spatial-temporal features in the multichannel data matrices. It also concatenates the outputs of the two convolutional-layer streams and learns a speed-volume feature vector.
- (3) The authors propose a new loss function for the deep learning model. It is the sum of a speed term and a weighted volume term. By appropriately setting the weight, the volume term improves the learning ability of the model and helps prevent overfitting.
- (4) Traditional studies on multi-lane traffic flow mostly focus on the mathematically modeling



3



Figure 1. The data input modeling process of converting the multi-lane traffic flow raw data to the multi-channel spatial-temporal matrix.

and behavior description of multi-lane traffic. This study is among the first efforts to apply deep learning methods for multi-lane traffic pattern mining and prediction.

Methodology

Modeling Multi-Lane Traffic as Multi-Channel Matrices

The first step of the methodology is modeling the multilane traffic flow as multi-channel matrices. The authors propose a data conversion method to convert the raw data into spatial-temporal multi-channel matrices, in which traffic on every individual lane is added to the matrices as a separate channel. This modeling idea comes from CNN's superiority to capture features in multichannel RGB images. In RGB images, each color channel has correlations vet differences with the other two. This is similar to traffic flows on different lanes in which correlations and differences both exist (32, 37). Thus, averaging traffic flow parameters at a certain milepost and timestamp is like doing a weighted average of the RGB values to get the grayscale value. In this sense, previous methods for traffic speed prediction are designed for "grayscale images" (spatial-temporal prediction for averaged speed) or even just a single image column (speed prediction for an individual location). In this study, the proposed model manages to handle lane-level traffic information by formulating the data inputs as "RGB images."

In this paper, loop detector data are used, as loop detectors collect different types of traffic flow data on individual lanes. That said, though the loop detector is a relatively traditional traffic detector, it provides lane-level traffic speed, volume, and occupancy data which many other detectors do not (42–44). For example, probe

vehicle data are widely used nowadays, but besides a small sample of traffic speeds and trajectories, most of them are unable to collect lane-level data or volume data.

This data conversion method diagram is shown in Figure 1. There are loop detectors installed at k different mileposts along this segment, and the past n time steps are considered in the prediction task. The number of lanes is denoted as c. Without loss of generality, it is assumed that the number of lanes is three in Figure 1 for the sake of illustration. Single-lane traffic would be represented by two $k \times n$ spatial-temporal 2D matrices, in which one is for speed and another for volume. They are denoted as I_u for speed and I_q for volume. The speed value and volume value are defined to be u_{ilt} and q_{ilt} , respectively, for a detector at milepost i (i = 1, 2, ..., k) and lane l (l = 1, 2, ..., c) at time t (t = 1, 2, ..., n). Note that each u_{ilt} or q_{ilt} is normalized to between 0 and 1 using min-max normalization as speed and volume have different value ranges. Thus, in the speed and volume matrices with the size $k \times n \times c$, the matrices are constructed using Equations 1 and 2,

$$I_u(i,t) = (u_{i1t}, u_{i2t}, \dots, u_{ict})$$
 (1)

$$I_q(i,t) = (q_{i1t}, q_{i2t}, \dots, q_{ict})$$
 (2)

where *i* and *t* are the row index and column index of a spatial-temporal matrix, representing the milepost and the timestamp, respectively. $I_u(i, t)$ and $I_q(i, t)$ denote the multi-channel pixel values of the speed and the volume. The number of channels corresponds to the number of lanes *c*. Each element in the 2D multi-channel matrices is a *c*-unit vector representing *c* lanes' traffic speeds or volumes at a given milepost *i* and time *t*. In the three-lane example in Figure 1, the spatial-temporal matrices have three channels. Mathematically, the spatial-temporal

multi-channel matrices for traffic speed (X_u) and volume (X_q) can be denoted as

$$X_{u} = \begin{bmatrix} I_{u}(1,1) & I_{u}(1,2) & \dots & I_{u}(1,n) \\ I_{u}(2,1) & I_{u}(2,2) & \dots & I_{u}(2,n) \\ \vdots & \vdots & & \vdots \\ I_{u}(k,1) & I_{u}(k,2) & \dots & I_{u}(k,n) \end{bmatrix}$$
(3)

$$X_{q} = \begin{bmatrix} I_{q}(1,1) & I_{q}(1,2) & \dots & I_{q}(1,n) \\ I_{q}(2,1) & I_{q}(2,2) & \dots & I_{q}(2,n) \\ \vdots & \vdots & & \vdots \\ I_{q}(k,1) & I_{q}(k,2) & \dots & I_{q}(k,n) \end{bmatrix}$$
(4)

Convolution for Feature Extraction

The CNN has demonstrated promising performance in image classification and many other applications as a result of its locally connected layers and the better ability than other neural networks to capture local features. In transportation, the traffic stream, as well as the disturbance to traffic stream, moves along the spatial axis and the temporal axis. Thus, applying CNN to the spatial–temporal traffic image manages to capture local features in both spatial and temporal dimensions. The fundamental operation in the feature extraction process of CNN is convolution. With the re-organized input as a multichannel matrix X (X could be X_u or X_q), the basic unit of a convolution operation is shown in Figure 2. On the left most of the figure, it is the input spatial–temporal matrix

or image X. Every channel of the input matrix is a 2D spatial-temporal matrix representing the traffic flow pattern on the corresponding lane. On the top of the leftmost column, channel #1 displays the traffic pattern of lane #1; and on the bottom, the pattern of lane #c is presented. The symbol "*" denotes the convolution operation in Figure 2. As the input is a multi-channel image, the convolution filters are also multi-channel. In the figure, a $3 \times 3 \times c$ filter is drawn, although the size of the filter can be changed in practice. The values inside the cells of a filter are weights of the CNN, which are automatically modified during the training process. The final weights are able to extract the most salient features in the multi-channel image. The convolution operation outputs a feature map for each channel, and they are summed up to be the extracted feature map of this convolution filter in the current convolutional layer. With multiple filters operated on the same input image, a multi-channel feature map will be constructed, and serves as the input to the next layer.

The TM-CNN for Speed Prediction

To learn the multi-lane traffic flow patterns and predict traffic speeds, a CNN structure is designed (see Figure 3). Compared with a standard CNN, the proposed CNN architecture is modified in the following aspects:

(1) The network inputs are different, that is, the input image is a spatial-temporal image built by



Figure 2. The convolution operation to extract features from the multi-channel spatial-temporal traffic flow matrices.



Figure 3. The proposed two-stream multi-channel convolutional neural network (TM-CNN) architecture.

traffic sensor data, and it has multiple channels that represent the lanes of a corridor. Moreover, the pixels' value range is different from a normal image. For a normal image, it is 0–255; however, here it ranges from 0 to either the highest speed (often the speed limit) or the highest volume (often the capacity).

- (2)The neural network has two streams of convolutional layers, which are for processing the speeds and volumes. But most CNN have only one stream of convolutional layers. The purpose of having two streams of convolutional layers is to integrate both speed information and volume information into the model so that the network can learn the traffic patterns better than only learning speed. To combine the two streams, a fusion operation that flattens and concatenates the outputs of the two streams is implemented between the convolutional layers and the FC layers. The fusion operation is chosen to be concatenation instead of addition or multiplying because the concatenation operation is more flexible for modifying each stream's structures. In other words, the concatenation fusion method allows the two streams of convolutional layers to have different structures.
- (3) The extracted features have unique meanings and are different from image classifications or most other tasks. The extracted features here are relations among road segments, time series, adjacent lanes, and between traffic flow speeds and volumes.

- (4) The output is different, that is, the output is a vector of traffic speeds of multiple locations at the future time rather than a single category label or some bounding boxes' coordinates. The output itself is part of the input for another prediction, whereas this is not the case for most other CNNs.
- (5) Different from most CNNs, the proposed CNN does not have a pooling layer. The main reason for not inserting pooling layers in between convolutional layers is that the input images are much smaller than regular images for image classification or object detection (45, 46). Regular input images to a CNN usually have hundreds of columns and rows, whereas the spatial-temporal images for roadway traffic are not that large. In this research and many existing traffic prediction studies, the time resolution of the data is 5 min, which means even using 2-h data for prediction there are only 24 time steps. Thus, there is no risk of losing information by pooling.
- (6) The loss function is devised to contain both speed and volume information. For traditional image classification CNNs, the loss function is the cross-entropy loss. And for traffic speed prediction tasks, the loss function is commonly the Mean Squared Error (MSE) function with only speed values. However, in this research, a new term in the loss function is added to incorporate the volume information. The ground truth speed vector and volume vector are denoted as as Y_u and Y_q , and the predicted speed vector and

volume vector as \hat{Y}_u and \hat{Y}_q . Note that Y_u , Y_q , \hat{Y}_u , and \hat{Y}_q are all normalized between 0 and 1. The loss function *L* is defined in Equation 5 by summing up the MSEs of speed and volume. The volume term $\lambda || \hat{Y}_q - Y_q ||_2^2$ is added to the loss function for reducing the probability of overfitting by helping the model better understand the essential traffic patterns. This design improves the speed prediction accuracy on the test dataset with proper settings of λ . The suggested value of λ is between 0 and 1 considering that the volume term that deals with overfitting should still have a lower impact than the speed term on speed prediction problems.

$$L = \left| \left| \hat{Y}_{u} - Y_{u} \right| \right|_{2}^{2} + \lambda \left| \left| \hat{Y}_{q} - Y_{q} \right| \right|_{2}^{2}$$
(5)

In the proposed TM-CNN, the inputs are the multichannel matrices X_u and X_q with the dimension of $k \times n \times c$. The filter size is all $2 \times 2 \times c$ to better capture the correlations between each pair of adjacent loops as well as adjacent times. The number of filters for each convolutional layer is chosen based on experience and the consideration to balance efficiency and accuracy. The last convolutional layer in each of the two streams is flattened and connected to a FC layer. This FC layer is fully connected with the output layer as well. The length of the output vector \hat{Y}_{u} is $1 \times (k \times c)$, as the prediction is for one future step. All activations except the output layer use the Relu function. The output layer has a linear activation function, which is adopted for regression tasks. Equations 6 and 7 describe the derivations mathematically from inputs to the outputs of the last convolutional layers,

$$\hat{Y}_{u}^{\text{conv}} = \phi \{ W_{u3} * \phi [W_{u2} * \phi (W_{u1} * X_{u} + b_{u1}) + b_{u2}] + b_{u3} \}$$

$$\hat{Y}_{q}^{\text{conv}} = \phi \{ W_{q3} * \phi [W_{q2} * \phi (W_{q1} * X_{q} + b_{q1}) + b_{q2}] + b_{q3} \}$$
(6)

where $\hat{Y}_{u}^{\text{conv}}$ and $\hat{Y}_{q}^{\text{conv}}$ are the intermediate speed and volume outputs of the CNN in between the last convolutional layers and the flatten layers, W_{ui} and W_{qi} (i = 1, 2, 3) are the weights for the convolutions, b_{ui} and b_{qi} (i = 1, 2, 3) are the biases, and $\varphi(\cdot)$ is the Relu activation function. After getting these two intermediate outputs, they are flattened and fused into one vector, and then further the relations are learned between the volume feature map and the speed feature map using FC layers.

As aforementioned, concatenation was chosen as the fusion function for the two flattened intermediate outputs to allow the customization of different neural network designs of the two streams. Customized streams could result in two intermediate outputs of different dimensions. Although concatenation would still successfully fuse the two outputs together and support the learning of speed–volume relationships by the FC layers, most other fusion operations require the two vectors to have the same length. This fusion process is mathematically represented in Equation 8 as follows,

$$\hat{Y}_{u}, \hat{Y}_{q} = W_{5} \times \varphi$$

$$\left[W_{4} \times \operatorname{Conc}\left(F(\hat{Y}_{u}^{\operatorname{conv}}), F(\hat{Y}_{q}^{\operatorname{conv}})\right) + b_{4}\right] + b_{5}$$
(8)

where W_4 and W_5 are the weights for the two FC layers, b_4 and b_5 are the biases, $F(\cdot)$ is the flatten function, and Conc(\cdot) is the concatenation function.

Case Study

Data Description

In this paper, 1-year loop data from January 1st, 2016 to December 31st, 2016 for a four-lane freeway corridor in Seattle are used for validation. Seattle is currently in the top ten of heaviest traffic cities in the United States, and this study freeway segment is one of the busiest corridors in Seattle. It starts from milepost-170 to milepost-165 of Interstate-5 (I5) freeway southbound, connecting the University of Washington to Downtown Seattle. There are 40 loop detectors on this corridor. They collect speed, volume, and occupancy traffic data. This study uses speed and volume for speed prediction. Occupancy is not included in the model because adding occupancy increases the training time and complexity yet does not improve the prediction accuracy. This can be explained by traffic flow theory: In most cases, if two of the three traffic parameters are known, the third one can be estimated. Thus, essentially, using three of them does not add more information to the prediction model. The data are downloaded from a traffic big data platform named Digital Roadway Interactive Visualization and Evaluation Network (DRIVE Net) (47), in which the data are aggregated to every 5 min. Data cleaning for missing values and outliers is done using a flexible and robust algorithm (48). Based on the data conversion method, speed and volume are each converted to a fourchannel matrix. The data conversion of the 1-year data generates about 105,000 data samples for model validation.

Model Implementation

(7)

The proposed model is implemented in Keras deep learning library using TensorFlow backend on an Nvidia GTX 1080 GPU. The implemented model architecture in the case study is shown in Figure 4. This architecture figure is automatically generated by Keras after the model



Figure 4. The model architecture diagram of the TM-CNN implemented in the case study. *Note*: TM-CNN = two-stream multi-channel convolutional neural network.

design. It displays the overall model structure as well as the dimensions of the inputs and outputs of all layers. Each input data sample is organized into the dimension of $10 \times 8 \times 4$, in which 10 is the number of detectors along the freeway segment on each lane, 8 is the number of time steps used for learning and prediction, and 4 is the number of channels (lanes). Here 8 was chosen as the number of time steps because it was observed that 8 is large enough (40 min) to ensure the model adequately captures the past traffic patterns, and short past time steps are used for efficient training. In this model, three convolutional layers are added sequentially to each stream. Three is selected as the number of hidden convolutional layers based on a trial-and-error process, during

Figure 5. Model parameter tuning for learning rate and the λ in the loss function.

which it was observed that three convolutional layers constantly outperform just having one or two convolutional layers, yet little improvement is observed with more than three. Two dropout layers are added to the model to reduce the probability of overfitting in the training process. One dropout layer is inserted between the last convolutional layer and the concatenation layer, with a dropout ratio 0.5; another is inserted between the FC layer (which is shown as a dense layer in Keras) and the output layer with a dropout ratio 0.25.

In the model validation process, the dataset is split into 80,000 samples for training and 25,000 samples for testing. In addition to the model architecture parameters, several other hyper-parameters need to be tuned in the training process. For example, the optimizer in the model training is RMSprop, given its faster convergence rate than other optimizers. One key hyper-parameter that often influences the performance of deep learning models is the learning rate, which determines how much the weights are adjusted with respect to the loss function gradient. It affects both model accuracy and training speed. In this case, the authors examined different learning rates ranging from 0.01 to 0.00001, and found that the learning rate around 0.0001 generated the best model accuracy. As shown in the first plot in Figure 5, training loss curves are plotted for different learning rates in the first 50 epochs. It can be observed that when the learning rate is smaller than 0.0001, smaller learning rate generates a smaller loss. However, when the learning rate is larger than 0.0001, the model loss starts becoming larger again, and at the same time, the model training takes a longer time. Thus, 0.0001 was chosen as the learning rate for this model.

Another critical parameter of the proposed model is the λ in the loss function. It determines the impact of traffic volume on the speed prediction. As aforementioned, the suggested value of λ is between 0 and 1 with the consideration that the volume term should have a

lower impact than the speed term. Therefore, ten values of λ were tested from 0 to 0.9 with an interval of 0.1. The curve of speed prediction accuracy on the test dataset with respect to λ is shown as the second plot in Figure 5. Compared with no volume term in the loss function, the speed prediction accuracy improves when $\lambda = 0.1$, which implies that the loss function design is effective. The model accuracy starts to decrease as λ getting larger from 0.1. This interesting finding indicates that: On the one hand, the volume term does have an impact on the speed prediction accuracy; on the other hand, the impact of the volume term should exist but not too large. This observation is reasonable: First, according to traffic flow theory, two traffic flow parameters can better determine the actual traffic flow status than just one parameter; second, as volume has more randomness and variation than speed in the short term, large impact of volume could increase the uncertainty in speed prediction.

Experimental Results and Comparison

To demonstrate the superiority of the proposed model, two evaluations were conducted. On the one hand, it was compared with five baseline models, on the other hand, both the ground truth speeds and the predicted speeds were visualized in the formats of spatial-temporal heat map and single-detector speed plot. ARIMA is one of the pioneering methods for traffic prediction; SVR is a popular model in the field before the large-scale applications of deep learning in traffic prediction; ANN is the traditional fully connected artificial neural network, which often serves as one baseline; LSTM, a specific type of recurrent neural network, is the most widely used model in recent years for traffic prediction. The proposed twostream CNN was also compared with a single-stream CNN, which merely contains a speed stream in the





Figure 6. (a) Heat maps showing the ground truth speeds (upper) and the predicted speeds (lower) for all four lanes from 6 a.m. to 8 p.m. on a day and (b) the predicted speeds and ground truths at milepost 166.4 for all lanes in 24 h.

network structure and no volume term in the loss function. The models were all fine-tuned to have their best performances. There are many other speed prediction models, some of which could be more advanced than the baseline models for specific tasks. However, considering that most of the existing models are not designed for multi-lane traffic pattern prediction, modifying them to predict multi-lane traffic just for comparison purposes could downgrade their capability and is also not meaningful at this point.

Table 1 shows the accuracies and comparison results. For each model, three different prediction time steps, that is, $5 \min$, $10 \min$, and $15 \min$ are examined. In general, the shorter the prediction time step is, the higher the

Table I. Accuracy Comparison with Baseline Methods

| | Prediction time steps | | |
|-------------------|-----------------------|------------|------------|
| | I (5 min) | 2 (10 min) | 3 (15 min) |
| ARIMA | 83.13% | 81.06% | 78.35% |
| SVR | 82.66% | 80.90% | 78.47% |
| ANN | 87.74% | 85.79% | 83.90% |
| LSTM | 88.46% | 86.78% | 84.65% |
| Single-stream CNN | 90.83% | 89.25% | 86.94% |
| TM-CNN | 91.21% | 90.06% | 88.15% |

Note: ARIMA = Auto Regressive Integrated Moving Average; SVR = Support Vector Regression; ANN = Artificial Neural Network; LSTM = Long Short-Term Memory; CNN = Convolutional Neural Network; TM-CNN = Two-Stream Multi-Channel Convolutional Neural Network. accuracy. This is consistent with most previous studies. It can be seen that the proposed two-stream multi-channel CNN has the best prediction accuracy over the baseline models in all three cases. The single-stream CNN in general beats the other four baseline models, but has a lower accuracy than the two-stream CNN. Also, it can be observed that with the increase of time step, the prediction accuracy differences between TM-CNN and other models generally become larger. These comparisons show three strengths of the proposed model: First, the conversion of raw traffic data to the multi-channel matrix indeed improves the learning and prediction ability by

better capturing spatial-temporal correlations between adjacent lanes, mileposts, and times. Second, the fusion of volume and speed further enhances the learning ability and model accuracy. Third, compared with the baseline methods, the TM-CNN demonstrates a better performance overall and its superiority in relatively longer-term speed prediction.

This study examined the influences of learning time steps and the number of locations. When the learning time steps are less than 30 min, the prediction accuracy increases as the time steps increase. However, when the time steps are larger than 30 min, the prediction accuracy does not change. This is also why 40 min was chosen as the learning time step. There was not much difference in accuracy among a different number of stations. In addition, the model training time increase as the number of stations or learning time steps increase.

Figure 6 shows visualization samples of the system accuracy. Figure 6a displays the heat maps of the ground truth speeds (upper) and the predicted speeds (lower) for every lane from 6 a.m. to 8 p.m. on a day. The prediction time step is 5 min in this visualization. The horizontal axis is the index of time. The vertical axis is the index for loop detectors, in which loops 0-9 are on lane #1 (the shoulder lane), and loops 30–39 are on lane #4 (the median lane). Figure 6b shows the speed prediction curves (the orange curves) and the ground truth curves (the blue curves) for single loops on every lane at milepost 166.4 for 24 h. Three observations can be summarized based on the visualizations in Figure 6, a and b: First, the proposed model achieves excellent learning and prediction performances in different traffic conditions (free flow and congestion). Second, the proposed model can learn and capture similar trends yet unique patterns of the traffic flow speed on all individual lanes. Third, the predicted speed values are smoother than the ground truth values. This is because of the variation and noise in real-world traffic flow and traffic data collection. The smoothness of the prediction actually demonstrates the ability of the proposed model to capture the general trends of traffic flow and its robustness to noise.

Conclusion

In this paper, the authors proposed a novel deep learning model called TM-CNN for multi-lane traffic speed prediction. Several new components were carefully designed and incorporated in the model to enable the effective learning of multi-lane traffic flow characteristics and the accurate prediction of multi-lane speeds. The new components included a raw data conversion method, a twostream multi-channel convolutional neural network architecture, and a new loss function.

Some interesting findings and recommendations can be concluded. (1) Experimental results demonstrate that the TM-CNN can learn and capture the traffic patterns in different traffic conditions and individual lanes. (2) Comparisons with the baseline models show that the TM-CNN achieves superior prediction accuracy and robustness over ARIMA, SVR, ANN, LSTM, and single-stream CNN. (3) The learning rate in the training process and the weight of the volume term in the loss function are critical hyper-parameters in this model. (4) For multi-lane traffic learning and prediction, the authors suggest converting traffic flow data into multichannel matrices using the proposed data conversion method. (5) The authors suggest incorporating traffic volume data into both the neural network architecture and the loss function for speed prediction tasks.

Future work will be carried out in two directions. First, this study conducted an initial experiment on a relatively small-scale dataset for the purpose of validating the model performance on predicting multi-lane traffic. Future studies will fine-tune and test the model for network-scale multi-lane traffic speed prediction. The second future direction is to modify the model structure to integrate data from ramp detectors into speed prediction. By doing this, the authors aim to further improve the learning ability and prediction accuracy of the model.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and system design: Ruimin Ke, Wan Li; data collection: Zhiyong Cui, Yinhai Wang; analysis and interpretation of results: Ruimin Ke, Wan Li, Zhiyong Cui; draft manuscript preparation: Ruimin Ke, Wan Li, Yinhai Wang. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was partially supported by the Pacific Northwest Transportation Consortium (PacTrans), U.S. DOT Region 10 University Transportation Center.

Data Accessibility Statement

The data used in this work are available on the Digital Roadway Interactive Visualization and Evaluation Network (DRIVE Net) platform at http://uwdrive.net/STARLab.

References

- Bennett, C. R. A Speed Prediction Model for Rural Two-Lane Highways. FHWA-RD-99-171. Federal Highway Administration, McLean, VA, 1994.
- Dougherty, M. S., and M. R. Cobbett. Short-Term Inter-Urban Traffic Forecasts Using Neural Networks. *International Journal of Forecasting*, Vol. 13, No. 1, 1997, pp. 21–31.
- Chen, H., and S. Grant-Muller. Use of Sequential Learning for Short-Term Traffic Flow Forecasting. *Transportation Research Part C: Emerging Technologies*, Vol. 9C, No. 5, 2001, pp. 319–336. https://doi.org/10.1016/S0968-090X (00)00039-5.
- Theja, P. V. V. K., and L. Vanajakshi. Short Term Prediction of Traffic Parameters Using Support Vector Machines Technique. *Proc., 3rd International Conference on Emerging Trends in Engineering and Technology*, Goa, India, 2010.
- Kumar, S. V., and L. Vanajakshi. Short-Term Traffic Flow Prediction Using Seasonal ARIMA Model with Limited Input Data. *European Transport Research Review*, Vol. 7, No. 21, 2015. https://doi.org/10.1007/s12544-015-0170-8.
- Clark, S. Traffic Prediction Using Multivariate Nonparametric Regression. *Journal of Transportation Engineering*, Vol. 129, No. 2, 2003, pp. 161–168.
- Kumar, S. V. Traffic Flow Prediction Using Kalman Filtering Technique. *Procedia Engineering*, Vol. 187, 2017, pp. 582–587.
- Vanajakshi, L., and L. R. Rilett. A Comparison of the Performance of Artificial. Neural Networks and Support Vector Machines for the Prediction of Traffic Speed. *Proc.*, *IEEE Intelligent Vehicles Symposium*, Parma, Italy, 2004.
- Chandra, S. R., and H. Al-Deek. Predictions of Freeway Traffic Speeds and Volumes Using Vector Autoregressive Models. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, Vol. 13, No. 2, 2009. https://doi.org/10.1080/15472450902858368.
- Smith, B. L., B. M. Williams, and R. Keith Oswald. Comparison of Parametric and Nonparametric Models for Traffic Flow Forecasting. *Transportation Research Part C: Emerging Technologies*, Vol. 10. No. 4, 2002, pp. 303–321. https://doi.org/10.1016/S0968-090X(02)00009-8.
- Liu, L., and K. Hall. A Hybrid Short-Term Traffic Speed Forecasting Model through Empirical Mode Decomposition (EMD) and Au- Toregressive Integrated Moving Average (ARIMA). Presented at 93rd Annual Meeting of the Transportation Research Board, Washington, D.C., 2014.

- Asif, M. T., J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. M. Dhanya, N. Mitrovic, and P. Jaillet. Spatiotemporal Patterns in Large-Scale Traffic Speed Prediction. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 15, No. 2, 2014, pp. 794–804.
- 13. Lecun, Y., Y. Bengio, and G. Hinton. Deep Learning. *Nature*, Vol. 521, 2015, pp. 436–444.
- 14. Ahmad, J., H. Farman, and Z. Jan. *Deep Learning Methods and Applications*. In *SpringerBriefs in Computer Science*, Springer, Singapore, 2019.
- Ke, R., Z. Li, J. Tang, Z. Pan, and Y. Wang. Real-Time Traffic Flow Parameter Estimation from UAV Video Based on Ensemble Classifier and Optical Flow. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, No. 1, 2019, pp. 54–64. https://doi.org/10.1109/TITS. 2018.2797697.
- Lee, E. M., J. H. Kim, and W. S. Yoon. Traffic Speed Prediction Under Weekday, Time, and Neighboring Links' Speed: Back Propagation Neural Network Approach. In Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues (D. S. Huang, L. Heutte, and M. Loog, eds.), Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, Vol. 4681, 2007.
- Jia, Y., J. Wu, and Y. Du. Traffic Speed Prediction Using Deep Learning Method. *Proc.*, 19th International Conference on Intelligent Transportation Systems, Rio de Janeiro, Brazil, 2016.
- Huang, S. H., and B. Ran. An Application of Neural Network on Traffic Speed Prediction under Adverse Weather Condition. University of Wisconsin–Madison, 2003.
- Ma, X., Z. Tao, Y. Wang, H. Yu, and Y. Wang. Long Short-Term Memory Neural Network for Traffic Speed Prediction Using Remote Microwave Sensor Data. *Transportation Research Part C: Emerging Technologies*, Vol. 54, 2015, pp. 187–197. https://doi.org/10.1016/j.trc.2015. 03.014.
- Tang, J., F. Liu, Y. Zou, W. Zhang, and Y. Wang. An Improved Fuzzy Neural Network for Traffic Speed Prediction Considering Periodic Characteristic. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, No. 9, 2017, pp. 2340–2350. https://doi.org/10.1109/TITS.2016. 2643005.
- Lv, Y., Y. Duan, W. Kang, Z. Li, and F. Y. Wang. Traffic Flow Prediction with Big Data: A Deep Learning Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 2, 2015, pp. 865–873. https://doi.org/ 10.1109/TITS.2014.2345663.
- Ma, X., Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang. Learning Traffic as Images: A Deep Convolutional Neural Network for Large-Scale Transportation Network Speed Prediction. *Sensors*, Vol. 17, No. 4, 2017, p. 818. https:// doi.org/10.3390/s17040818.
- Cui, Z., R. Ke, and Y. Wang. Deep Bidirectional and Unidirectional LSTM Recurrent Neural Network for Network-Wide Traffic Speed Prediction. arXiv preprint arXiv:1801.02143, 2018.
- 24. Cui, Z., K. Henrickson, R. Ke, and Y. Wang. High-Order Graph Convolutional Recurrent Neural Network: A Deep

Learning Framework for Network-Scale Traffic Learning and Forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 2018. https://doi.org/10.1109/ TITS.2019.2950416.

- Zhang, Z., M. Li, X. Lin, Y. Wang, and F. He. Multistep Speed Prediction on Traffic Networks: A Graph Convolutional Sequence-to-Sequence Learning Approach with Attention Mechanism. arXiv preprint arXiv:1810.10237, 2018.
- Wang, J., R. Chen, and Z. He. Traffic Speed Prediction for Urban Transportation Network: A Path Based Deep Learning Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 100, 2019, pp. 372–385.
- Yu, H., Z. Wu, S. Wang, Y. Wang, and X. Ma. Spatiotemporal Recurrent Convolutional Networks for Traffic Prediction in Transportation Networks. *Sensors*, Vol. 17, No. 7, 2017, p. 1501.
- Yu, B., H. Yin, and Z. Zhu. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. arXiv preprint arXiv:1709.04875, 2017.
- 29. Yao, H., X. Tang, H. Wei, G. Zheng, Y. Yu, and Z. Li. Modeling Spatial-Temporal Dynamics for Traffic Prediction. *arXiv preprint arXiv:1803.01254*, 2018.
- Ma, X., Y. Li, Z. Cui, and Y. Wang. Forecasting Transportation Network Speed Using Deep Capsule Networks with Nested LSTM Models. arXiv preprint arXiv:1811.04745, 2018.
- Ahn, S., and M. J. Cassidy. Freeway Traffic Oscillations and Vehicle Lane-Change Maneuvers. *Transportation and Traffic Theory*, Vol. 1, 2007, pp. 691–710.
- Daganzo, C. F. A Behavioral Theory of Multi-Lane Traffic Flow. Part II: Merges and the Onset of Congestion. *Transportation Research Part B: Methodological*, Vol. 36, No. 2, 2002, pp. 159–169.
- Michalopoulos, P. G., D. E. Beskos, and Y. Yamauchi. Multilane Traffic Flow Dynamics: Some Macroscopic Considerations. *Transportation Research Part B: Methodological*, Vol. 18, No. 4–5, 1984, pp. 377–395.
- 34. Shiomi, Y., T. Taniguchi, N. Uno, H. Shimamoto, and T. Nakamura. Multilane First-Order Traffic Flow Model with Endogenous Representation of Lane-Flow Equilibrium. *Transportation Research Part C: Emerging Technologies*, Vol. 59, 2015, pp. 198–215.
- Wagner, P., K. Nagel, and D. E. Wolf. Realistic Multi-Lane Traffic Rules for Cellular Automata. *Physica A: Statistical Mechanics and its Applications*, Vol. 234, No. 3, 1997, pp. 687–698. https://doi.org/https://doi.org/10.1016/ S0378-4371(96)00308-1.
- Klar, A., and R. Wegener. A Hierarchy of Models for Multilane Vehicular Traffic I: Modeling. *SIAM Journal on Applied Mathematics*, Vol. 59, No. 3, 1998, pp. 983–1001.
- Daganzo, C. F. A Behavioral Theory of Multi-Lane Traffic Flow. Part I: Long Homogeneous Freeway Sections.

Transportation Research Part B: Methodological, Vol. 36, No. 2, 2002, pp. 131–158. https://doi.org/10.1016/S0191-2615(00)00042-4.

- Shvetsov, V., and D. Helbing. Macroscopic Dynamics of Multilane Traffic. *Physical Review E*, Vol. 59, No. 6, 1999, p. 6328.
- Tang, J., F. Liu, W. Zhang, R. Ke, and Y. Zou. Lane-Changes Prediction Based on Adaptive Fuzzy Neural Network. *Expert Systems with Applications*, Vol. 91, 2018, pp. 452–463. https://doi.org/10.1016/j.eswa.2017.09.025.
- Ke, R., Z. Li, S. Kim, J. Ash, Z. Cui, and Y. Wang. Real-Time Bidirectional Traffic Flow Parameter Estimation from Aerial Videos. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, No. 4, 2017, pp. 890–901.
- Liu, Q., B. Wang, and Y. Zhu. Short-Term Traffic Speed Forecasting Based on Attention Convolutional Neural Network for Arterials. *Computer-Aided Civil and Infrastructure Engineering*, Vol 33, No. 11, 2018, pp. 999–1016. https:// doi.org/10.1111/mice.12417.
- Ke, R., Z. Zeng, Z. Pu, and Y. Wang. New Framework for Automatic Identification and Quantification of Freeway Bottlenecks Based on Wavelet Analysis. *Journal of Transportation Engineering Part A: Systems*, Vol. 144, No. 9, 2018, pp. 1–10. https://doi.org/10.1061/JTEPBS.0000168.
- Feng, S., R. Ke, X. Wang, Y. Zhang, and L. Li. Traffic Flow Data Compression Considering Burst Components. *IET Intelligent Transport Systems*, Vol. 11, No. 9, 2017, pp. 572–580.
- Zhuang, Y., R. Ke, and Y. Wang. Innovative Method for Traffic Data Imputation Based on Convolutional Neural Network. *IET Intelligent Transport Systems*, Vol. 3, No. 4, 2018, pp. 605–613.
- Chen, X., S. Wang, C. Shi, H. Wu, J. Zhao, and J. Fu. Robust Ship Tracking via Multi-View Learning and Sparse Representation. *Journal of Navigation*, Vol. 72, No. 1, 2019, pp. 176–192. https://doi.org/10.1017/S03734633 18000504.
- 46. Ke, R., J. Lutin, J. Spears, and Y. Wang. A Cost-Effective Framework for Automated Vehicle-Pedestrian Near-Miss Detection Through Onboard Monocular Vision. *Proc.*, *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Honolulu, Hawaii, 2017, pp. 25–32.
- Wang, Y., W. Zhang, K. Henrickson, R. Ke, and Z. Cui. Digital Roadway Interactive Visualization and Evaluation Network Applications to WSDOT Operational Data Usage. WA-RD 854.1. Washington State Department of Transportation, 2016.
- Henrickson, K., Y. Zou, and Y. Wang. Flexible and Robust Method for Missing Loop Detector Data Imputation. *Transportation Research Record: Journal of the Transportation Research Board*, 2015. 2527: 29–36.