TOWARDS PREDICTING TRAFFIC SHOCKWAVE FORMATION AND PROPAGATION: A CONVOLUTIONAL ENCODER-DECODER NETWORK

Mohammadreza Khajeh Hosseini
Ph.D. Student
Texas A&M University
199 Spence St. College Station, TX 77840
Email: mreza@tamu.edu
Tel: +1 682 208 4187

Alireza Talebpour, Ph.D., Corresponding Author
Assistant Professor
Texas A&M University
199 Spence St. College Station, TX 77840
Email: atalebpour@tamu.edu
Tel: +1 979 845 0875

Word Count: 3828 words + 2 table(s) $\times$ 250 = 4328 words

Submission Date: July 24, 2019

Submitted for presentation at the 99th Annual Meeting of the Transportation Research Board, and publication in Transportation Research Record

ABSTRACT

This study proposes a deep learning methodology to predict the propagation of traffic shockwaves. The input to the deep neural network is time-space diagram of the study segment, and the output of the network is the predicted (future) propagation of the shockwave on the study segment in the form of time-space diagram. The main feature of the proposed methodology is the ability to extract the features embedded in the time-space diagram to predict the propagation of traffic shockwaves.

Keywords: Shockwave Propagation, Traffic Prediction, Convolutional Encoder-Decoder

INTRODUCTION AND BACKGROUND

The boundary between two different traffic states is known as traffic shockwave. The driving dynamics change from one state to another as the speed of the vehicles and their spacing changes. The differences between the two traffic states can be mild such as a high-speed traffic stream reaching a traffic stream with moderate speed and density, or it can be significant when reaching a high density and low-speed traffic stream (e.g., congested area). In general, when the traffic state changes, the vehicles need to respond by adjusting their speed and acceleration. Currently, the approaches adopted for guidance (e.g., lane changing) of the autonomous vehicles involves the consideration of the current state of the surrounding vehicles in terms of their location and speed with limited attention to the response of the other vehicles to their surrounding environment and how the traffic state could evolve (e.g. formation of shockwaves). As a result, predicting the propagation of traffic shockwave in time and space can help in improving both the safety and performance of the autonomous vehicles. Considering the valuable information that the connected vehicles could provide, we are proposing a methodology to predict the propagation of traffic shockwaves that accounts for both the individual behaviors and the collective change in the state of the traffic.

The state of traffic is characterized by density, flow, and speed that change over time and space (i.e., along the roadway). Lint and Hinsbergen (*1*) classify traffic prediction methodologies into three general groups of naive, parametric, and non-parametric approaches. Naive approaches have no data-driven model parameters, and they are generally established on the average of the historical observations or the prediction based on the current state (*2*). Parametric methodologies apply to the traffic prediction models that utilize a traffic flow model with parameters estimated based on the historical data or jointly with new observations. The fundamental diagram, along with first and second order traffic flow models are examples of the well-researched parametric models relating the macroscopic characteristic of traffic state (*3, 4*). Balancing between accuracy and model complexity to address irregularities and time changing dynamics is one of the challenges of adopting a parametric traffic prediction methodology. Non-parametric refer to the methodologies that do not depend on traffic flow models and are build on simple data-driven techniques such as linear regression (*5*), neural networks (*6*), support vector regression (*7*), as well as time series analysis (*8*). Adopting the data-driven approaches for traffic prediction has become more popular with the increase in the availability of the data resources in the past decades. Most of the existing non-parametric models use macroscopic data such as flow, density, and speed for traffic state prediction, while few studies consider individual-level trajectories for prediction (*9, 10*). The impact of abrupt individual behavior on traffic state increases with the increase in flow and density levels. As a result, this study considers the microscopic level interactions for better prediction of the traffic state. For a more comprehensive review of traffic prediction models, interested readers can refer to (*11*).

The connected vehicles' technology provides the opportunity to disseminate useful data to share with the drivers, or to complement the vehicle's onboard sensors of autonomous vehicles (AV). The connected vehicles share safety-related data regarding their location and speed with other vehicles and traffic control centers to improve their safety and to optimize their travel time and quality. The connectivity provides the opportunity to monitor the traffic stream and how the traffic evolves over time and space more extensively than the location specific traffic monitoring devices. The individual level location data transmitted by connected vehicles help to construct the time-space diagram. The time-space diagram is a comprehensive representation of the traffic stream without any abstraction or aggregation. The traffic flow dynamics as well as vehicles'

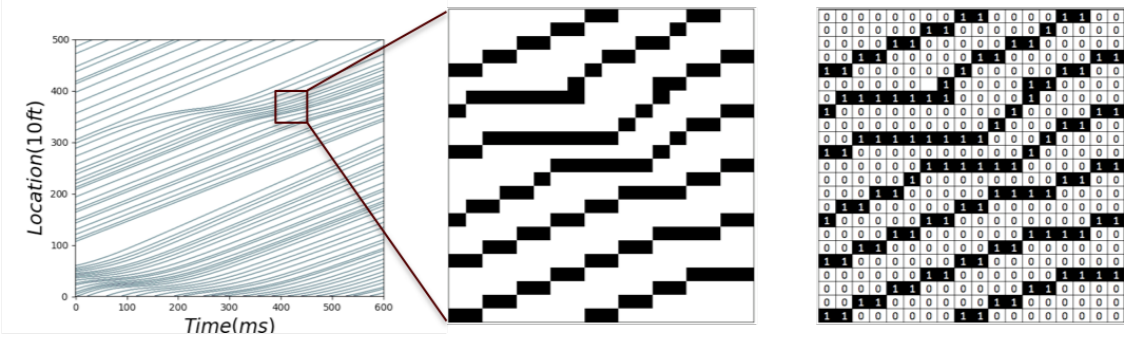
interactions and shockwave formation and propagation are embedded in the time-space diagram. Khajeh-Hosseini and Talebpour (*10*) used a matrix representation of the time-space diagram to predict the future traffic state of the roadway segment in the form of average flow and density. The objective of this study is to introduce a methodology to predict the propagation of traffic shockwaves (i.e., the change of flow and density over time and space) using the similar matrix representation of the time-space diagram. The main innovation of the proposed methodology is introducing the model that can learn the features embedded in the time-space diagram to predict the propagation of the traffic shockwaves.

METHODOLOGY

The interaction between the vehicles and the traffic state is best captured in the time-space diagram. The time-space diagram is the plot of the trajectory of the vehicles traversing the roadway (i.e., space domain) over time. This plot is comprehensive and provides the location and speed (slope of each trajectory) of the vehicles, as well as the interaction between the vehicles and the traffic state. Besides, the traffic shockwaves, which are the boundary between the different traffic states are evident in the time-space diagram. As a result, this study adopts the time-space diagram as a valuable input and predicts the propagation of shockwave in a time-space diagram format.

Time-Space Diagram

In order to construct the time-space diagram as the input for the prediction, this study assumes a connected vehicles environment. In this environment, the connected vehicles share their speed and location every 0.1 second comparable to the basic safety message (BSM (*12*)) transmitted through wireless communication. The time-space diagram can be generated in the form of a time-space matrix as proposed by Khajeh-Hosseini and Talebpour (*10*). The time-space matrix, figure 1 approximates the time-space diagram by dividing the time and space domains into bins of 10 ft by 100 ms (discretization). The time-space matrix is a binary matrix, and the rows represent the discrete space domain, and the columns represent the discrete time domain. In this matrix, the cell value of one indicates the presence of a vehicle in that space and time bin, and the value of zero indicates an empty bin. The time-space matrix is a 2D tensor that can be used in the convolution process.



a) Time-space diagram b) Time-space bins c) Time-space matrix

FIGURE 1 Time-Space Diagram (*10*)

Convolutional Encoder-Decoder

This study proposes the use of a deep neural network to predict the propagation of the traffic shockwave from the current time-space diagram of the vehicles as shown in figure 2. The convolutional encoder-decoder structure is an appealing type of mapping function for this study as the input and output of the networks are 2D tensors with similar properties. The convolution is the process of sliding a fixed size filter (e.g., three-dimensional receptive window) over the input tensor. Each convolutional layer applies the convolution process to the output of the previous layer and provides an output tensor. The convolution process accounts for the spatial correlation between the units that fit in the receptive window of the filter. Also sliding the same filter over the input space ensures feature detection independent of its location. This location independence aspect of the convolutional layers makes them a practical choice to encode the time-space matrix since the traffic shockwaves can occur at any point in time and space.

There are different convolutional encoder-decoder network architectures depending on the use of convolutional, deconvolutional, pooling and upsampling layers. Some networks only use convolutional layers in both encoder and decoder components such as the Fully Convolutional Network (FCN) (*13*) and Seg-Net (*14*). While other networks such as DeconvNet (*15*) and RED-Net (*16*) use deconvolutional layers in the decoder component. Some of the challenges in the convolutional encoder-decoder network is the vanishing gradient, and reconstructing lost features from the max-pooling and convolution process. The use of skip connections (*16*), and memorizing the maximum features (*14, 15*) of the pooling process to use for the upsampling are the solutions. The skip connections inspired by the residual network (ResNet)(*17*) allow the signal to be propagated to the bottom layers and address the vanishing gradient.

The proposed encoder-decoder architecture in figure 2 is inspired by the RED-Net (*16*) developed for image restoration, and consists of symmetric layers of convolution and deconvolution with skip-layer connections. The encoder component of the network contains three pairs of convolutional layers with the small receptive window of 3×3 and increasing channels from 16 to 64. The decoder component of the network contains symmetrical deconvolutional layers. The deconvolution process, unlike the convolution process, associates a single input to multiple outputs. The encoder component encodes the features embedded in the time-space diagram. The decoder component predicts the propagation of the traffic shockwaves in the form of a new time-space diagram.

The skip-layers connect the symmetric convolution and deconvolution layers every two layers. The skip-layer connection sums the convolutional feature maps with the deconvolutional feature map element-wise. The encoding convolutional layers extract the main features and abstracts the input, while the deconvolution layers decode the abstract input and predict the shockwave propagation. The proposed network is deep, and the skip-layer connections provide the opportunity to propagate the gradient to the beginning layers of the network. The skip-layer connections address the vanishing gradient problem in very deep networks. Moreover, the proposed architecture is capable of taking any size of time-space matrix as input since the network only utilizes the convolutional and deconvolutional layers.

Data

Simulation

There are limited available data on the trajectory of the individual vehicles to create an extensive data set for the training of the proposed deep neural network that can generalize well. As a result, to

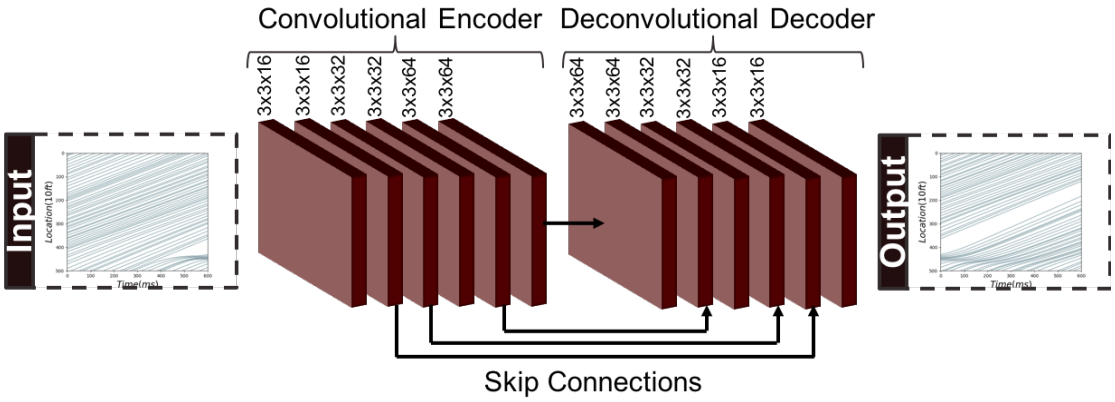


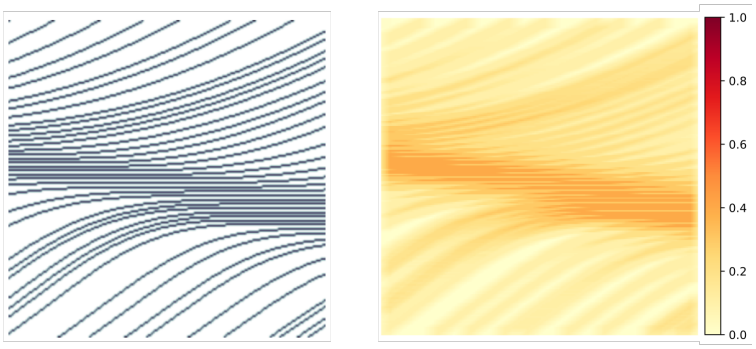
FIGURE 2 Shockwave Prediction: a convolutional encoder-decoder approach.

construct a comprehensive and large data set, this study uses a microscopic simulator written in the Python programming language to collect the trajectory of the vehicles. The microscopic simulator adopts the Intelligent Driver Model (IDM) (*18*) as its car-following logic, and the MOBIL (*19*) as its lane-changing logic.

The simulation collects the trajectory of the vehicles traversing a three-lane roadway segment with the length of 40,000 feet over 15 minutes. At every simulation runs, unique and random IDM and MOBIL parameters are assigned to every vehicle to make the simulation more realistic. In order to create a data set with different traffic states from free-flow to fully congested, two types of disturbances are used in the simulation. Sudden deceleration of a random vehicle for a small period (e.g., 15 seconds) creates a speed drop perturbation and disturbs the traffic stream. Another type of disturbance used in the simulation is forcing a random slow-moving vehicle for a more extended period, such as 5 minutes to create congestion and traffic breakdown. Both of the disturbances result in the formation of shockwaves in the traffic stream. The start time of these two disturbances are limited during periods of $(20i, 20i + 20)$ seconds, and i being even numbers between 0 and 45. When constructing pairs of input and output data for the training of the model, this limitation becomes useful. The model cannot predict random occurrences of these disturbances, and this limitation helps to exclude the start of these disturbances from the output data. Also, for each simulation run, the desired speed is randomly (uniform distribution) selected from different speed limits including 30, 45, 50, 55, 65, 70, 75 mph to create a more comprehensive data set.

Input and output data

The proposed encoder-decoder (figure 2) of this study takes the time-space matrix as input and predicts the propagation of the traffic shockwaves in the same time-space matrix form. The binary time-space matrix (figure 1) not only presents the traffic shockwaves but also depicts the crisp location of the individual vehicles in time and space domains. Training the network to output a binary tensor of shape $(200, 200)$ is challenging, and breaking the binary constraint improves the training. Averaging the cells of the time-space matrix with their neighbors, figure 3a, blurs the exact location of the vehicles on time and space domains; however, averaging over a small window maintains the propagation of traffic shockwaves (figure 3b). The colors of points on the averaged time-space diagram presented in figure 3b change from light yellow to red proportional to the value of cell ranging from 0 to 1. Taking the averaged time-space matrix as the type of output



a) Time-space diagram b) Averaged time-space diagram

FIGURE 3 Averaged time-space diagram

improves the training of the network. The encoder-decoder network approximates the mapping function from the averaged time-space matrix of segment x over the period of $(t-20,t)$ to the averaged time-space matrix of the segment x over the period of $(t,t+20)$. The averaged time-space matrix is derived by replacing every cell in the time-space matrix with the average of itself and its neighbors up to 100 feet and 1 second (i.e., averaging window of 100 ft by 1 s).

Training data

The microscopic traffic simulator provides a 40000 ft by 900 s time-space diagram for each lane and every simulation run. This diagram can be divided into 900 smaller time-space diagrams for segments of 2000 ft and a shorter period of 20 s. The 900 smaller time-space diagrams are divided into 450 pairs of input and output data. One could extract more pairs of input and output if relax the limitation on keeping the start of the artificial disturbances in the input data. This study collected data from more than 2000 simulation runs resulting in more than 0.9 million data points. The collected data is divided into three groups of training, validation, and testing sets with ratios of 80%, 10%, and 10% respectively.

Training

Training is the iterative process of adjusting the trainable parameters of the model to gradually minimize the loss function. The convolutional encoder-decoder of this study contains 180,449 trainable parameters. Adopting the small receptive window of 3×3, and fully convolutional and deconvolutional layers kept the number of network parameters small. The model parameters are updated in multiple iterations (steps). At each iteration, the loss function is estimated for a batch of data points, and the parameters are adjusted based on their loss gradient times the learning rate (a small constant). The Adam optimizer (*20*) is a stochastic gradient-based optimizer that is adapted for the training of the network of this study.

Loss Function

The prediction model of this study is a regression model that maps the current averaged time-space diagram to the future averaged time-space diagram. The mean squared error (MSE), equation (1), is a standard performance measure used as the loss function for the training of regression type neural networks. The output of network (model) F with parameters Θ for input X^i is $F(X^i;\Theta)$, and the true value of output is Y^i. Mean absolute error (MAE), equation (2), is another performance measure

for regression problems. However, the MAE is not useful as the loss function and estimation of gradients in neural networks.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} ||F(X^i;\Theta) - Y^i||^2 \quad (1)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} ||F(X^i;\Theta) - Y^i|| \quad (2)$$

The input and output of the model of this study are 2D tensors of size $(200, 200)$. A smoothed version of the output can be constructed by replacing each cell of the output tensor with the average of itself and its neighboring cells. A well trained neural network model is expected to predict outputs very comparable to the true outputs. Besides, it is expected that the smoothed versions of the predicted and true outputs are also comparable. In order to speed up the training (convergence) of the model and to guide the gradients, this study proposes the use of the custom loss function of equation (3). The MSE_{10}, MSE_5, and MSE_3 are the estimated MSE between the true and predicted outputs when smoothed with sliding average windows of size 10×10, 3×3, and 5×5 respectively. The size of the sliding window indicates the extent of neighboring cells considered in the estimation of the average for that cell. Adopting this custom loss function significantly improved the convergence of the training process.

$$loss = MSE + 1000(MSE_{10} + MSE_5 + MSE_3) \quad (3)$$

The training process of the model is conducted in two steps. In the first step, the model is trained using the loss function of equation 3 until the loss value on the validation set started increasing. In the second step, the model is retrained using the MSE, equation (1), as the loss function.

RESULTS AND DISCUSSIONS ON RESULTS

In the training process of the model, a batch size of 60 and the early stopping policy are used to prevent overfitting the training data. The loss function is estimated at the end of every epoch (a complete iteration over the entire dataset), and the training is stopped after five epochs from the one with the minimum loss on the validation dataset. Table 1 presents the performance of the network of this study in prediction on the validation and testing dataset. Training the model with the custom loss function, equation (3), helped the convergence in the first step of training. Also, retraining the model in the second step by adopting the original MSE as the loss function further improved the performance of the model from MSE error of 0.0037 to 0.0029. According to table 1, the performance of the fully trained model in terms of MSE and MAE on the testing dataset are 0.0030 and 0.0408 respectively.

Traffic shockwave propagation prediction

Figure 4 presents some of the traffic predictions of the model in the form of the averaged time-space matrix. The prediction model of this study takes the averaged time-space matrix of time $(t-20, t)$ as input (x), and predicts the future averaged time-space matrix of time $(t, t+20)$ as the output (y). Comparing the predictions of the model (predicted y) and the true states of the traffic (y) in figure 4, the model is capable of predicting the propagation of the traffic shockwaves. According to this figure, the predicted averaged time-space diagrams presents dissemination, propagation, the forward and backward movement of the traffic shockwaves over the evaluated segment of roadway.

TABLE 1 Model performance on time-space matrix

Validation Dataset		
	MSE	$MSE_{10} + MSE_5 + MSE_3$
Training Step 1	0.0037	0.0071
Training Step 2	0.0029	NA
Testing Dataset		
	MSE	*MAE*
Trained Model	0.0030	0.0408

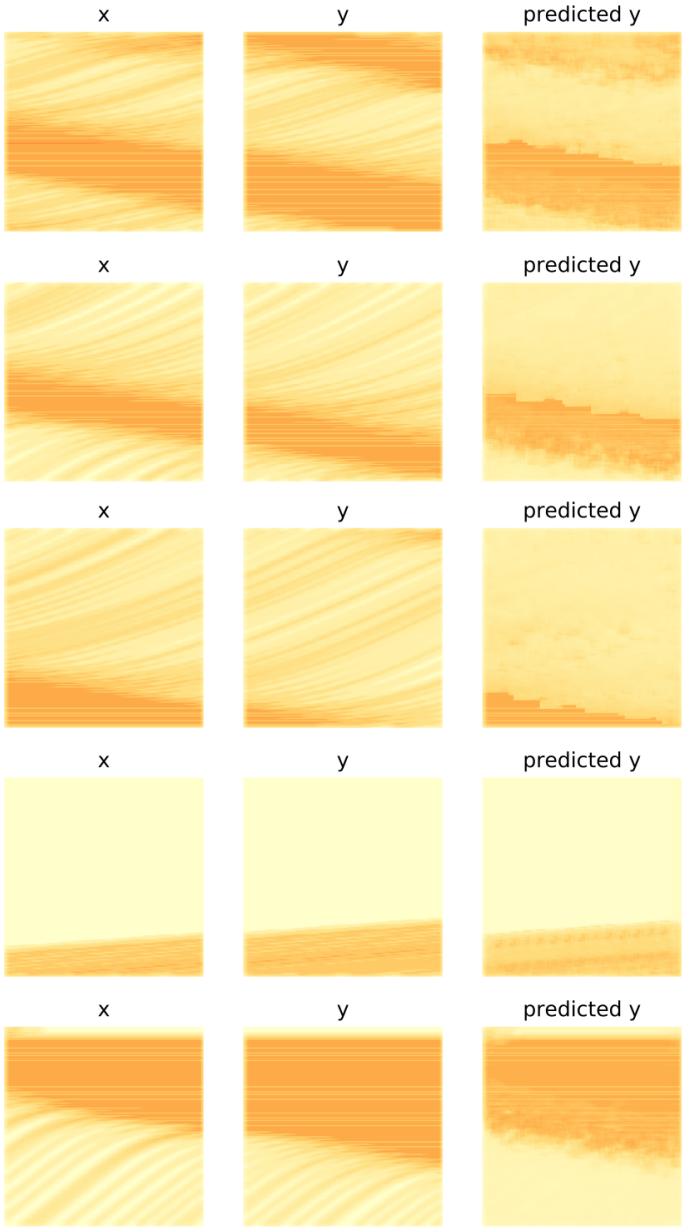


FIGURE 4 Traffic shockwave propagation prediction results

Density time-space matrix

The traffic shockwave propagation prediction of the network can be evaluated more quantitatively. The traffic shockwave is the boundary between two states of the traffic. Edie (*21*) estimates the average density $k(A)$ for a time-space block of A (e.g. 100 feet by 1 second) based on equation (4). In this equation, $|A|$ is the area of the time-space block A, and $t(A)$ stand for the total time spent by all the vehicles going through block A.

$$k(A) = \frac{t(A)}{|A|} \tag{4}$$

As specified in the methodology section, the time-space matrix is a binary matrix constructed by dividing time and space domains into bins of 10 ft by 100 ms. In this matrix, one represents the presence of a vehicle in that time and space bin, and zero represents an empty bin. The number of occupied bins of the time-space block A is equal to the summation of all the bins of its representative binary time-space matrix (i.e., $sum(A)$). As a result, the total time spent by all the vehicles going through any arbitrary time-space block of A is equal to multiplying the number of occupied bins in that block by 0.1 second, equation 5.

$$t(A) = 0.1 \times sum(A) \tag{5}$$

Considering Edie's (*21*) definition on the average density of a time-space block, the averaged time-space matrix ($\overline{TS}$) can be used to estimate the density time-space matrix (K). Similar to the time-space matrix, the rows of this matrix represent the discrete space domain, and their columns represent the discrete time domain. The values of each cell in the matrix K is the average density of a time-space block (e.g., 100 ft by 1 s) centered at that location in time and space. The density time-space matrix depicts the change in traffic state over time and space and consequently the propagation of the traffic shockwaves.

As mentioned in the methodology section, the averaged time-space matrix ($\overline{TS}$) of this study is estimated by replacing every cell in the time-space matrix with the average of itself and its neighbors up to 100 ft and 1 s. Each cell in the time-space matrix is representative of a bin with dimensions of 10 ft in space domain and 0.1 s in the time domain. The averaging window of 100 ft by 1 s is equivalent to a 10×10 averaging window on the time-space matrix, in other words, each cell of the averaged time-space matrix ($\overline{TS}$) is the average of 100 cells in the time-space matrix. Therefore, if the averaged time-space matrix ($\overline{TS}$) is multiplied by 100, the cells of the resulting matrix indicate the number of occupied cells in the blocks of 100 ft by 1 s centered on that location on the time-space matrix. As a result, the density time-space matrix (K) can be estimated based on equation (6). In this equation, the constant 5280 is applied for the unit conversion from feet to mile. Vehicles per mile (vpm) is the unit for the values in the resulting density time-space matrix (K).

$$K = \frac{t(A)}{|A|} = \frac{100 \times \overline{TS} \times 0.1}{100 \times 1} \times 5280 = 528 \times \overline{TS} \tag{6}$$

According to equation (6), the averaged time-space matrix ($\overline{TS}$) can be converted to the density time-space matrix (K) by a constant scalar. Therefore, the prediction (output) of the model is proportional to the density time-space matrix. The performance of the model in table 1 are updated for the density time-space matrix presented in table 2. According to table 2, the mean absolute error of the model in predicting the density for small blocks of 100 ft by 1 s is 21.54 vpm.

TABLE 2 Network performance on density time-space matrix

Validation Dataset		
	MSE	$MSE_{10} + MSE_5 + MSE_3$
Training Step 1	1031.50	1979.36
Training Step 2	808.47	NA
Testing Dataset		
	MSE	*MAE*
Trained Model	836.35	21.54

Root mean squared error (RMSE) is another valuable performance measure that has the same unit as the output. Based on table 2, the RMSE of the model in the prediction of the density on the testing dataset is estimated as 28.91 vpm. Considering a range of 200 vpm for the density, the MAE and RMSE of the model are between %10 to %14 of the range of density.

CONCLUSION

This study proposes a methodology to predict the propagation of traffic shockwaves in the form of the averaged time-space matrix. The averaged time-space matrix is comparable to a density time-space matrix derived from Edie's definition of average density. The traffic shockwave is the boundary between two traffic states, and the density time-space matrix depicts the state changes in the form of density. The result of the analysis indicated that the model is capable of predicting the dissemination, propagation, the forward and backward movement of the traffic shockwaves over the study segment. Moreover, the performance of the model in the form of MAE and RMSE for predicting the density time-space matrix is 21.54 vpm and 28.91 vpm respectively. Considering a range of 0 to 200 vpm for the density, the performance of the model is acceptable for prediction of the traffic shockwaves propagation.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 1826410.

AUTHOR CONTRIBUTION

All authors contributed to all aspects of the study from conception and design, to data collection, to analysis and interpretation of results, and manuscript preparation. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Van Lint, J. and C. Van Hinsbergen, Short-term traffic and travel time prediction models. *Artificial Intelligence Applications to Critical Transportation Issues*, Vol. 22, No. 1, 2012, pp. 22–41.
2. Eglese, R., W. Maden, and A. Slater, A road timetableTM to aid vehicle routing and scheduling. *Computers & operations research*, Vol. 33, No. 12, 2006, pp. 3508–3519.
3. Zheng, Z. and D. Su, Traffic state estimation through compressed sensing and Markov random field. *Transportation Research Part B: Methodological*, Vol. 91, 2016, pp. 525–554.
4. Wang, R., D. B. Work, and R. Sowers, Multiple Model Particle Filter for Traffic Estimation and Incident Detection. *IEEE Trans. Intelligent Transportation Systems*, Vol. 17, No. 12, 2016, pp. 3461–3470.
5. Wilby, M. R., J. J. V. Díaz, A. B. Rodríguez Gonz´ lez, and M. Á. Sotelo, Lightweight occupancy estimation on freeways using extended floating car data. *Journal of Intelligent Transportation Systems*, Vol. 18, No. 2, 2014, pp. 149–163.
6. Polson, N. G. and V. O. Sokolov, Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, Vol. 79, 2017, pp. 1–17.
7. Castro-Neto, M., Y.-S. Jeong, M.-K. Jeong, and L. D. Han, Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert systems with applications*, Vol. 36, No. 3, 2009, pp. 6164–6173.
8. 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. 3, 2015, p. 21.
9. Elfar, A., C. Xavier, A. Talebpour, and H. S. Mahmassani, *Traffic Shockwave Detection in a Connected Environment Using the Speed Distribution of Individual Vehicles*, 2018.
10. Khajeh-Hosseini, M. and A. Talebpour, TRAFFIC PREDICTION USING TIME-SPACE DIAGRAM: A CONVOLUTIONAL1NEURAL NETWORK APPROACH. *Transportation Research Board*, 2019.
11. Seo, T., A. M. Bayen, T. Kusakabe, and Y. Asakura, Traffic state estimation on highway: A comprehensive survey. *Annual Reviews in Control*, Vol. 43, 2017, pp. 128–151.
12. SAE, J2735 dedicated short range communications (dsrc) message set dictionary. *Society of Automotive Engineers, DSRC Tech Committee*, 2016.
13. Long, J., E. Shelhamer, and T. Darrell, Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
14. Badrinarayanan, V., A. Kendall, and R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 12, 2017, pp. 2481–2495.
15. Noh, H., S. Hong, and B. Han, Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1520–1528.
16. Mao, X., C. Shen, and Y.-B. Yang, Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in neural information processing systems*, 2016, pp. 2802–2810.

17. He, K., X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
18. Treiber, M., A. Hennecke, and D. Helbing, Congested traffic states in empirical observations and microscopic simulations. *Physical review E*, Vol. 62, No. 2, 2000, p. 1805.
19. Kesting, A., M. Treiber, and D. Helbing, General lane-changing model MOBIL for car-following models. *Transportation Research Record*, Vol. 1999, No. 1, 2007, pp. 86–94.
20. Kingma, D. P. and J. Ba, Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
21. Edie, L. C., Car-following and steady-state theory for noncongested traffic. *Operations research*, Vol. 9, No. 1, 1961, pp. 66–76.