

# Signalized Arterial Origin-destination Flow Estimation Using Flawed Vehicle Trajectories: A Self-supervised Learning Approach Without Ground Truth

*Qinzheng Wang<sup>a</sup>, Yun Yuan<sup>a</sup>, Qiwei Zhang<sup>b</sup>, Xianfeng Terry Yang<sup>b,\*</sup>*

*<sup>a</sup>Department of Civil & Environmental Engineering, University of Utah, Salt Lake City, UT 84112, USA*

*<sup>b</sup>Department of Civil & Environmental Engineering, University of Maryland, College Park, MD 20742, USA*

## Abstract

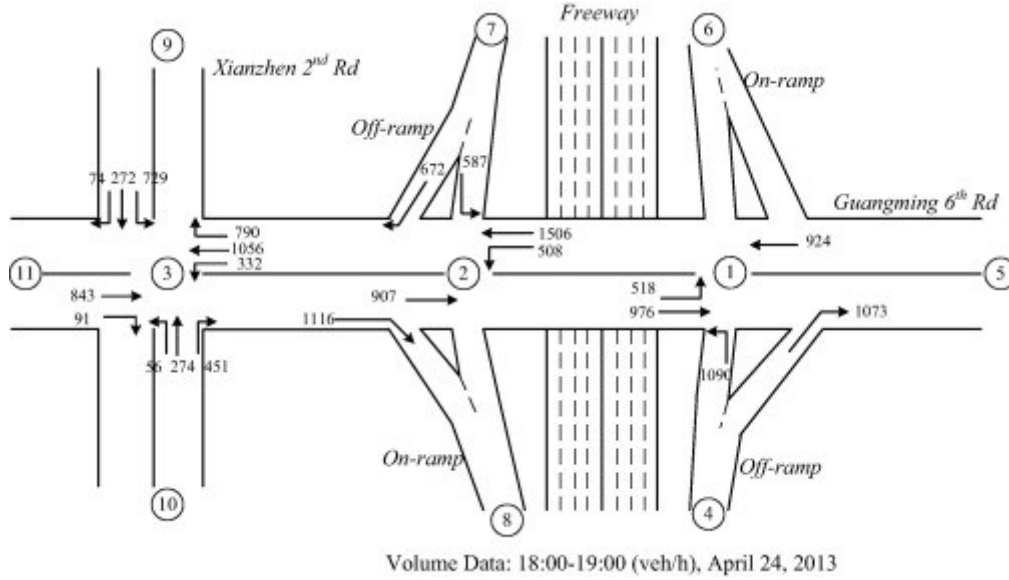
For alleviating arterial congestion, most control strategies provide progression for through and turning traffic. A prerequisite input is the arterial origin-destination (OD) flow pattern, which can be estimated based on connected vehicle (CV) trajectories. However, the existing estimation methods require the ground-truth historical OD flow, which is difficult to obtain. To address this issue, this paper develops a method to estimate real-time OD flow along a signalized arterial without ground truth. A model based on the Generative Adversarial Network (GAN) network is proposed, which incorporates long short-term memory (LSTM), attention mechanism, and convolutional neural network (CNN) to capture the temporal and spatial correlations between OD flow patterns. This model is trained with the proposed self-supervised without historical OD flow. The proposed model is extensively tested based on a realistic signalized arterial, and the results indicate sufficient accuracy for progression control.

**Keywords:** Signalized arterial, Origin-destination flow, trajectory, Generative Adversarial Network, self-supervised learning

## 1. Introduction

Origin-Destination (OD) flow estimation on arterials over space and time can achieve insights into traffic patterns and assists traffic management. For example, the conventional two-way progression frequently causes an overflow at turning bays, resulting in network gridlock. Considering this situation, a multi-path signal progression control system is investigated (Yang et al., 2015; Arsava et al., 2016; Wang et al., 2021). Some studies have demonstrated that providing progression to multiple path with heavy traffic volume can improve the arterial control efficiency and vehicle operational efficiency (Yang et al., 2015; Arsava et al., 2016;). In addition, Wang et al. (2021) proposed a traffic signal control system which incorporated adaptive traffic signal control at the intersection level and multi-path progression control at the corridor level. Their study has demonstrated that signal coordination was provided not only for through traffic but also other critical paths, which could improve the performance of the network. One of the key questions is how to identify real-time critical paths which

are defined as paths with high traffic flows. Therefore, the most fundamental and critical information for such a control system is the time-varying distribution of the origin-destination (OD) flow along the signalized arterial. To illustrate the arterial OD flow, Fig. 1 shows an arterial layout in Chupei, Taiwan, including three intersections, which connect the urban road network and a congested freeway. The numbers near the arrows represent the traffic volume of the movement. It can be observed that turning volumes from-or-to the on-ramps and off-ramps are heavy.



**Fig. 1. An illustrative example of an arterial segment in Chupei, Taiwan (Yang et al., 2015)**

Very limited studies have been conducted on estimating arterial OD flow patterns along signalized arterials. Conventionally, the arterial OD flow is estimated based on traffic counts according to several conservation laws. Specifically, the link flows should be the summation of all path flows traveling through the link (Chang and Wu, 1994; Lin, 2006), which can only be applied to a simple transportation network (Rao et al., 2018). Lou and Yin (2010) proposed a framework to dynamically estimate ODs along a signalized arterial by using link counts. They decomposed this problem into two sub-problems at the intersection level and the corridor level. They firstly inferred the turning movements at each intersection with link counts and then treated them as observations. However, their study did not consider the impact of the signal timing plan exerted on time-varying link flows, which consequently results in low accuracy. Yang and Chang (2015) proposed three models for estimating signalized arterial OD flow. The first model was based on the relationship between link counts and OD. The second model focused on modeling the interrelationships between OD flow patterns and intersection turning counts. Considering the impacts of traffic signal timing plans, the third model, incorporating time-varying queue length information, is proposed to increase the estimation accuracy. However, computing efficiency and tractability are challenging due to a large number of unknown parameters.

With the increasing data availability, data-driven OD flow estimation has been

investigated to address this problem, such as license plate recognition (LPR) data (Castillo et al., 2013; Chiou et al., 2011; Sun et al., 2011, Mo et al., 2020), automatic vehicle identification (AVI) data (Van Der Zijpp, 1997; Asakura et al., 2000; Dixon and Rilett, 2002; Antoniou et al., 2004; Dixon and Rilett, 2005; Zhou and Mahmassani, 2006; Chen et al., 2011; Hadavi and Shafahi, 2016; Cao et al., 2021), cellphone data (Sohn and Kim, 2008; Iqbal et al., 2014), and probe vehicle data (Matsumoto et al., 2005; Yamamoto et al., 2009; Asmundsdottir et al., 2010; Baek et al., 2010; Cao et al., 2013; Yang et al., 2017;). The basic idea of those methods is to boost estimation accuracy by supplementing information which is unavailable before. Despite the lack of arterial OD flow estimation, the previous studies on flow estimation on the road network can be categorized as maximum likelihood models (Geva et al., 1983; Irving et al., 1986; Cascetta and Nguyen, 1988; Spiess, 1987); Generalized least squares (GLS) model (Bell, 1983; Cascetta, 1984; Bell, 1991; Cascetta et al., 1993; Sherali and Park, 2001; Cascetta et al., 2013), Bayesian-based model (Geva et al., 1983; Maher, 1983; Hazelton, 2008; Perrakis et al., 2012; Wang and Mirchandani, 2013; Castillo et al., 2008b, 2014), state-space models (Okutani and Stephanedes, 1984; Ashok and Ben-Akiva, 2002; Zhou and Mahmassani, 2007; Alibabai and Mahmassani, 2008; Lu et al., 2015), and simple neural networks (Kikuchi et al., 1993; Yang et al., 1998; Gong, 1998; Mussone et al., 2010; Padinjarapat & Mathew, 2013; Remya and Mathew, 2013; Zhao et al., 2017). However, these methods are developed for network OD estimation. Most of those studies are traffic-assignment-based methods, which are based on several assumptions. For example, it assumed traffic information is available to all traffic users and they always select the shortest path, which is not in consistent with the reality. Besides, there is only one path for each OD pair along the arterial. Therefore, those existing methods may not be applicable for the signalized arterial OD estimation.

To leverage trajectory data in OD flow estimation, Ou et al. (2019) proposed a framework to dynamically estimate arterial OD flow with prior information, which is barely applicable. Huang et al. (2019) constructed a deep learning model based on Recurrent Neural Network to learn the evolution pattern of OD flow. This method requires that all links in the road network are equipped with camera detectors to capture vehicle information, which may result in high installation and maintenance costs. Considering the large dimensional difference between the observed traffic information and OD flow to be estimated, Tang et al. (2021) proposed a convolution-based deep neural network to learn the relationship between features extracted from AVI-based information and OD flow. This study applied transfer learning to mitigate the obstacle in obtaining ground-truth OD flow information, but still needs some prior OD flow information. In sum, three critical issues of the state-of-art data-driven methods lie in the literature: (a) Prior information and historical arterial OD flow are required to obtain acceptable results; (b) Recognition and identification systems are required to get traffic information (e.g., link counts), resulting in high installation and maintenance costs; (c) The malfunctions of detectors will contribute to serious estimation inaccuracies.

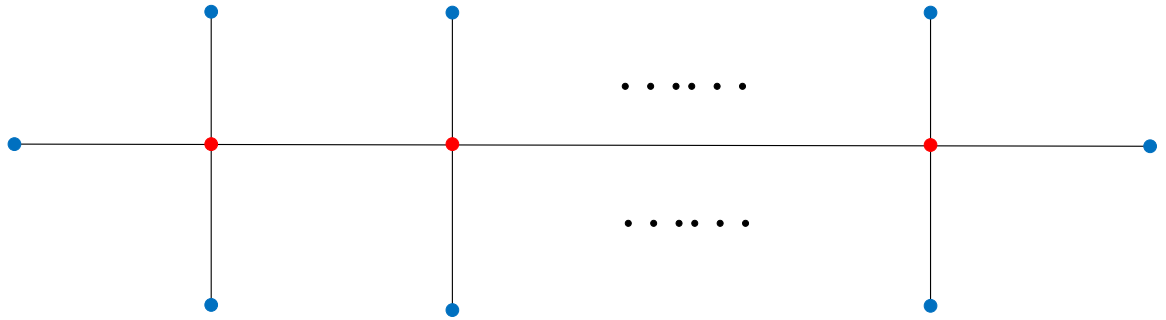
To address these issues, this paper first proposes a novel framework for learning real-time OD flow estimation without ground truth. A self-supervised learning method is proposed to leverage partial trajectory data from connected vehicle (CV) to estimate

the OD flow of both CV and regular vehicles. More specifically, we firstly obtain a partial OD matrix simply by aggregating the collected CV data. Then an original matrix is obtained by dividing the partial OD matrix by the global market penetration rate and such matrix is considered to be flawed. Second, this study customizes a generative adversarial network (GAN) model to repair the OD matrix. This study integrates the long short-term memory (LSTM), attention mechanism, and convolutional neural network (CNN) into the customized Generative Adversarial Network (GAN) model to capture both temporal and spatial patterns of OD flow. Third, the proposed self-supervised learning method trains the customized GAN for estimating arterial OD.

The remainder of the paper is organized as follows. Section 2 states the problem of signalized arterial OD flow estimation. Section 3 describes the training method without ground truth data and introduces the model to estimate OD flow. Section 4 conducts an experimental study and presents the results. The last section summarizes the conclusions and recommendations for future research directions.

## 2. Problem Statement

This research considers a signalized arterial with several intersections, represented by  $G = (N, L)$ , where  $N$  denotes the node set, specified by  $N = \{1, 2, \dots, n\}$ ;  $n$  represents the node where two roads intersect (i.e., the red nodes in Fig. 1) or the node that vehicles enter to and exit from the intersection (i.e., the blue nodes in Fig. 2).  $L$  denotes the link set in which each component is the link between two adjacent nodes, specified by  $L = \{1, 2, \dots, l\}$ . Blue nodes in Fig. 2 are considered as origin nodes and destination nodes. By connecting each origin node and destination node, a set of OD pairs, represented by  $I$ , can be constructed,  $I = \{1, 2, \dots, i\}$ . Let  $T$  denote the analysis period and dividing it into  $k$  uniform intervals, the set of time intervals is denoted by  $K = \{1, 2, \dots, k\}$ .



**Fig. 2. A simplified signalized arterial structure**

Based on the above notation definition, the traffic pattern along a signalized arterial during the analysis period can be represented by an OD matrix,  $M$ , as follows:

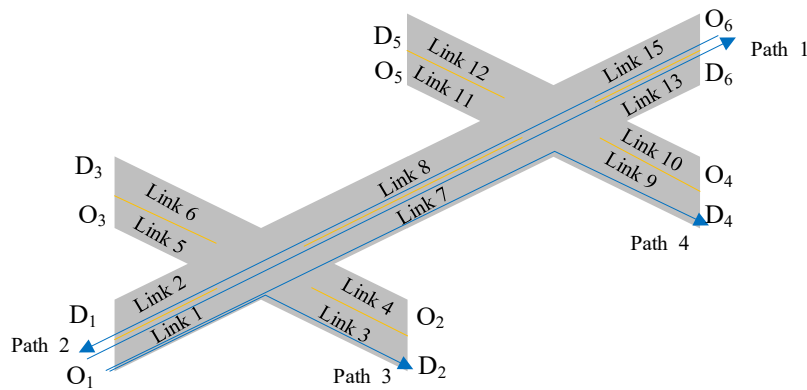
$$M = \begin{bmatrix} M_1^t \\ M_2^t \\ \vdots \\ M_i^t \end{bmatrix} = \begin{bmatrix} x_1^1 & x_1^2 & \cdot & \cdot & \cdot & x_1^K \\ x_2^1 & x_2^2 & \cdot & \cdot & \cdot & x_2^K \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_l^1 & x_l^2 & \cdot & \cdot & \cdot & x_l^K \end{bmatrix} \quad (1)$$

where  $x_i^k$  denotes the number of vehicles traveling between OD pair  $i$  within time interval  $k$ . The OD matrix  $M$  is the output of this problem.

This research assumes two types of vehicles, regular vehicles and CV, travel along this signalized arterial. Since CV broadcasts real-time locations, it is easy to obtain the number of CV traveling between each OD pair, namely CV OD matrix, within each time interval. The CV OD matrix  $M_C$  is the input of this problem:

$$M_C = \begin{bmatrix} M_{C_1}^t \\ M_{C_2}^t \\ \cdot \\ \cdot \\ \cdot \\ M_{C_l}^t \end{bmatrix} = \begin{bmatrix} x_{c_1}^1 & x_{c_1}^2 & \cdot & \cdot & \cdot & x_{c_1}^K \\ x_{c_2}^1 & x_{c_2}^2 & \cdot & \cdot & \cdot & x_{c_2}^K \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{c_l}^1 & x_{c_l}^2 & \cdot & \cdot & \cdot & x_{c_l}^K \end{bmatrix} \quad (2)$$

To further illustrate the problem, Fig. 3 shows the signalized arterial with 6 origins, 6 destinations, and 30 OD pairs. The path between each OD pair is composed of several links: (a) only partial paths have common links. For example, path 1 shares link 1 with path 3 while they do not have common links with path 2; (b) The number of shared links is different for paths, e.g., path 1 and path 3 only have one shared link, link 1, but path 1 and path 4 have two shared links, link 1 and link 7. For those paths with shared links, the turning directions of vehicles on shared links are uncertain, resulting in different path flows. This may contribute to complicated dependencies between different OD flows. Such relationships are highly time-dependent.



**Fig. 3. An illustrative case of a signalized arterial**

### 3. Methodology

#### 3.1 Notation

To help the understanding of the proposed model, the used notations are listed in the following table.

**Table 1. Notations**

|                             |                                                                           |
|-----------------------------|---------------------------------------------------------------------------|
| Arterial                    |                                                                           |
| $M$                         | Number of intersections                                                   |
| $G$                         | Signalized arterial structure                                             |
| $N$                         | Node set                                                                  |
| $L$                         | Link set                                                                  |
| $I$                         | Set of OD pairs                                                           |
| $T$                         | Analysis period                                                           |
| $K$                         | Set of time interval                                                      |
| $\mathcal{M}$               | OD matrix of all vehicles                                                 |
| $x_i^k$                     | Number of vehicles traveling between OD pair $i$ within time interval $k$ |
| $\mathcal{M}_c$             | OD matrix of CV                                                           |
| $x_{ci}^k$                  | Number of CV traveling between OD pair $i$ within time interval $k$       |
| General deep neural network |                                                                           |
| $x_i$                       | Inputs of deep neural network                                             |
| $y_i$                       | Ground truth data                                                         |
| $f_\theta$                  | Parametric vector                                                         |
| $n$                         | Number of training samples                                                |
| $L_R$                       | Empirical risk                                                            |
| J-invariant function        |                                                                           |
| $M^f$                       | Original OD samples                                                       |
| $M^c$                       | Ground truth OD matrix                                                    |
| $b$                         | Flaw                                                                      |
| $m$                         | Dimensions of each matrix                                                 |
| $\mathcal{J}$               | A given partition of dimensions of a matrix                               |
| $J$                         | Sub-dimensions of $\mathcal{J}$                                           |
| $x_J$                       | Values of a matrix on dimension $J$                                       |
| $f(u)_J$                    | Values of a function on dimension $J$                                     |
| $\mathcal{L}(f)$            | Self-supervised loss function                                             |
| $J^c$                       | Complement of $J$                                                         |
| $f_\theta$                  | Denoising function                                                        |
| $g_\theta$                  | Classical function                                                        |
| $h(u)$                      | Function replacing the values on the location $J$                         |
| $q(u)$                      | Neighborhood of an element in a matrix                                    |
| $l, w$                      | Random value of an element in a matrix                                    |

|                                        |                                                           |
|----------------------------------------|-----------------------------------------------------------|
| $r$                                    | Width of the surrounding window of an element in a matrix |
| Specific deep neural network structure |                                                           |
| $P(.), Q(.)$                           | Certain distribution                                      |
| $p(.), q(.)$                           | Probability function                                      |
| $G$                                    | Generator of GAN                                          |
| $D$                                    | Discriminator of GAN                                      |
| $\tau_i$                               | Input gate of LSTM                                        |
| $\tau_f$                               | Forget gate of LSTM                                       |
| $\tau_o$                               | Output gate of LSTM                                       |
| $\tilde{a}_t$                          | Candidate state of LSTM                                   |
| $h_t$                                  | State of step $t$ in LSTM                                 |
| $x_t$                                  | Input of step $t$ in LSTM                                 |
| $W_i, W_f, W_o, W_c$                   | Weights in LSTM                                           |
| $b_i, b_f, b_o, b_c$                   | Bias in LSTM                                              |
| $\sigma, \tau$                         | Activation functions                                      |
| $Q$                                    | Query vector                                              |
| $K$                                    | Key vector                                                |
| $V$                                    | Value vector                                              |
| $W^Q, W^K, W^V$                        | Weights of attention mechanism                            |
| $d_k$                                  | Dimension of the key vectors of attention mechanism       |
| $y_{conv}$                             | Output of convolutional layer                             |
| $b$                                    | Bias in convolutional neural network                      |

### 3.2 Novel Framework of the Self-supervised Learning Model

This research aims to mine the real-time trajectory data from  $M_C$  to estimate time-dependent OD flows of a signalized arterial (e.g., elements in  $M$ ). We can view this estimation problem from a probabilistic perspective. Considering the spatio-temporal relationship, each OD,  $x_i^k$ , can be viewed as a random variable depending on the collected CV OD flow of all OD pairs within historical time intervals and the conditional probability  $P(x_i^k | x_{c_1}^1: x_{c_i}^{k-1})$  follows a certain distribution. Here, we use  $x_{c_0}^0: x_{c_i}^{m-1}$  to indicate the historical CV records for all OD pairs. Then probabilistic inference methods can be adapted to solve the unknown distribution. Conventionally, the unknown distribution can be obtained by a model  $Q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}, \theta)$  controlled by several parameters  $\theta$  and then let this model approximate the real data distribution  $P(x_i^k | x_{c_1}^1: x_{c_i}^{k-1})$ , as shown in Eq. (3).

$$Q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}, \theta) \sim P(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}) \quad (3)$$

The difference between these two distributions is usually measured by Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951):

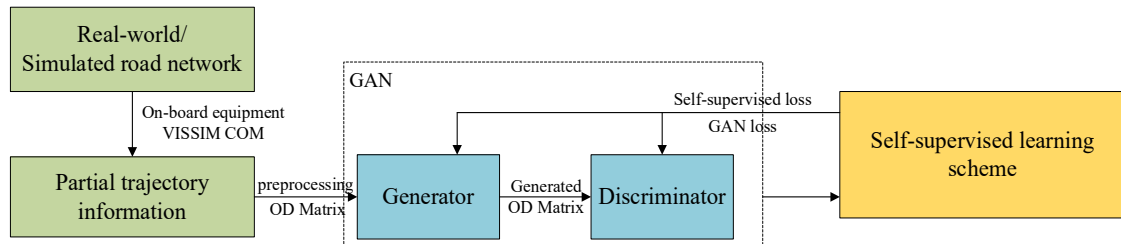
$$\begin{aligned} & \text{KL}(q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}) | p(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}, \theta)) = \\ & \int_{x_j^m} q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}) \log \frac{q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1})}{p(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}, \theta)} dx_i^k \end{aligned} \quad (4)$$

Then the optimal parameters can be obtained by minimizing the KL divergence, as shown in Eq. (5).

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \text{KL}(q(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}) || p(x_i^k | x_{c_1}^1: x_{c_i}^{k-1}, \theta)) \quad (5)$$

There are two challenges to solving the transformed divergence minimization problem. First, the concrete formulations of the two distributions are unknown. It is difficult to measure the divergence between them theoretically (Park et al., 2008; Sun et al., 2004). Second, previous methods require ground-truth data to get meaningful numerical output. To tackle the two problems, we customize GAN, which can train a generative machine that is associated with an implicit experimental distribution to capture the temporal and spatial relationship of ODs, and propose a self-supervised training model to train the GAN without ground truth.

Conventional GANs are under the supervised learning framework, where the ground-truth outputs are required. However, the historical OD flow is not accessible in this problem due to the low penetration rate of CV. To address this issue, this study aims to propose a self-supervised learning technique to training GAN without ground truth. The architecture of GAN is shown in Fig. 4 and the detailed network structures of the generator and discriminator are introduced in the following part. Fig. 4 shows the overall framework of arterial OD flow estimation. CVs that equipped with on-board units (OBU) can transmit vehicle trajectory information to roadside unit (RSU) in real-time. CV OD flow information can be achieved by processing trajectory data within each time interval. Based on this and considering the features of OD flows, we design two different models as the generator and the discriminator to capture the temporal and spatial correlations of OD flows. General approaches to train the generator and discriminator are based on fully observed OD flows. However, the only available information we can leverage is CV OD flow. Hence, a self-supervised training method only based on the CV OD flow is proposed.



**Fig. 4. Pipeline of the proposed self-supervised learning with no ground truth**

### 3.3 Learning Without Ground Truth Method

Unlike traditional methods that explicit a priori statistical modeling of estimation, recent advances in deep neural networks have sparked significant interest in mapping flawed observations to the unobserved variables. This happens by training a regression model with a large number of pairs  $(x_i, y_i)$  of inputs  $x_i$  and ground truth data  $y_i$  and minimizing the empirical risk

$$\operatorname{argmin}_{\theta} \sum_{i=1}^n L(f_{\theta}(x_i), y_i) \quad (6)$$

where  $f_{\theta}$  is a parametric vector, and  $n$  is the number of training samples.

The empirical risk can typically be defined as the type of mean square error (MSE) as follows:

$$L_R = \frac{1}{n} \sum_{i=1}^n (f_{\theta}(x_i) - y_i)^2 \quad (7)$$

Apart from MSE, other optimization functions, such as mean absolute error (MAE), mean bias error (MBE), can also be applied. Regardless of which function, ground truth data is essential. However, it is difficult to access the ground truth data (i.e., ODs along the signalized arterial) in this paper. Therefore, it is challenging to estimate without ground truth data by using the typical empirical risk. In this paper, we propose a new empirical risk.

To better elaborate our model, we treat the inputs to the model, the original OD matrix estimated from trajectories, as a matrix and each component (i.e., each estimated path flow) is estimated by dividing the CV OD flow by global CV market penetration rate (e.g., 20%). Let  $M^f$  represents the original OD matrix and we name it flawed matrix.  $M^c$  represents the ground truth OD matrix and we name it clear matrix. Then a flawed matrix can be represented by

$$M^f = M^c + b \quad (8)$$

where  $b$  denotes flaw.  $M^c, M^f \in R^m$ , where  $m = h * w * c$  depends on the spatial and channel dimensions.

Since the flaw matrix is obtained by dividing the CV OD flow matrix by the global market penetration rate, thus, we furthermore assume the flaw to be zero-mean

$$E(b_i) = 0 \quad (9)$$

Then

$$E(M_i^f) = M_i^c \quad (10)$$

Eq (9) and Eq (10) mean that if we obtain multiple flawed matrices of the same true matrix based on different flaws, the result shall reach the true matrix by averaging those flawed matrices.

Here, we borrow the concept of  $J$  – *invariant* function (Baston and Royer, 2019), as follows:

**Definition 1.** Consider a clean matrix  $M^c$  and its flawed measurement  $M^f$ ,  $M^c, M^f \in R^m$ , where  $m = h * w * c$  depends on the spatial and channel dimensions. For a given partition  $\mathcal{J} = \{1, 2, 3, \dots, m\}$  of the dimensions of a matrix  $u \in R^m$ . Let  $J \in \mathcal{J}$ . A function  $f: R^m \rightarrow R^m$  is  $J$  – *invariant* if  $f(M^f)_J$  does not depend on the value of  $x_J$ . where  $f(M^f)_J$  and  $x_J$  denote values of  $f(M^f)$  and  $u$  on  $J$ . For each  $J \in \mathcal{J}$ , if the function is  $J$  – *invariant*, then  $f$  is  $\mathcal{J}$  – *invariant*.

We use Fig. 5 to explain the specific meaning of  $J$  – *invariant* in the application of OD flow estimation. Fig. 5 is an example original OD matrix  $x$ .  $J$  is a subset of the matrix, and  $f$  is a  $J$  – *invariant* function. It has the property that the value of  $f(x)$  does not dependent on the value of  $x$  restricted to  $J$ ,  $x^J$ .

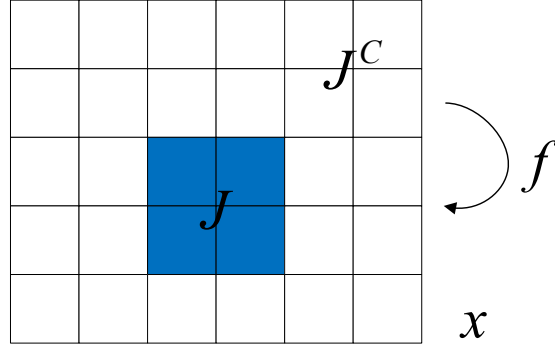


Fig. 5. An example OD flow matrix

Then we propose the self-supervised loss function over  $\mathcal{J}$  – *invariant* function  $f$  as follows:

$$\mathcal{L}(f) = E \| f(M^f) - M^f \|^2 \quad (11)$$

Based on this, consider a flawed matrix  $M^f \in R^m$  and its ground truth  $M^c \in R^m$ . Assume the flaw is zero mean and independent on the input data, the actual optimal estimator through any class of  $\mathcal{J}$  – *invariant* functions to estimate the true value from flawed data alone by minimizing the self-supervised loss function  $\mathcal{L}(f)$ .

$$\begin{aligned} E_{M^f} \| f(M^f) - M^f \|^2 &= E_{M^f, M^c} \| f(M^f) - M^c \|^2 + E_{M^f, M^c} \| M^f - M^c \|^2 - \\ &\quad 2 \langle f(M^f) - M^c, f(M^f) - M^c \rangle \end{aligned} \quad (12)$$

Since  $f$  is  $\mathcal{J}$  – *invariant*, then  $f(M^f)_J$  and  $M^f_J$  are independent variables for all  $J \in \mathcal{J}$ . Then the third term can be reduced to

$$\begin{aligned}
\langle f(M^f) - M^c, f(M^f) - M^c \rangle &= E_{M^c} \langle E_{M^f|S} [f(M^f) - M^c], E_{M^f|S} [M^f - M^c] \rangle \\
&= E_{M^c} (E_{M^f|M^c} [f(M^f) - M^c] E_{M^f|M^c} [M^f - M^c]) \quad (13)
\end{aligned}$$

Because the noise is zero mean, then

$$E_{M^c} (E_{M^f|M^c} [f(M^f) - M^c] E_{M^f|M^c} [M^f - M^c]) = 0 \quad (14)$$

Then

$$E_{M^f} \|f(M^f) - M^f\|^2 = E_{M^f, M^c} \|f(M^f) - M^c\|^2 + E_{M^f, M^c} \|M^f - M^c\|^2 \quad (15)$$

In the above equation, the first term is ground truth loss. The second term is noise variance, which is in particular independent on  $f$ . Therefore, when  $f$  is  $\mathcal{J}$ -invariant, minimizing the self-supervised loss  $E_{M^f} \|f(M^f) - M^f\|^2$  indirectly minimizes the supervised loss  $E_{M^f, M^c} \|f(M^f) - M^c\|^2$ .

According to definition 1, the output of  $f(x)_J$  in those  $J$  coordinates does not depend on its input in those coordinates. Intuitively,  $\mathcal{J}$ -invariant means  $f$  only uses the complement of  $J$ ,  $J^c$ , to repair  $x_J$ . Thus, any  $\mathcal{J}$ -invariant function can be treated as a set of functions  $f_J: R^{J^c} \rightarrow R^J$ . Then, the self-supervised loss function can be changed to

$$\mathcal{L}(f) = \sum_{J \in \mathcal{J}} E \|f_J(M^f_{J^c}) - M^f_J\|^2 \quad (16)$$

The  $\mathcal{J}$ -invariant function  $f_J^*$  minimizing Eq. (12) satisfies

$$f_J^*(M^f)_J = E(M^c_J | M^f_{J^c}) \quad (17)$$

for each  $J \in \mathcal{J}$ .

This means that the optimal  $\mathcal{J}$ -invariant estimator for  $M^c$  in dimensions  $J$  refers to their expected value conditional on  $M^f$  in the dimensions out of  $J$ . We aim to apply the aforementioned methods to obtain the optimal parameter  $\theta$  for the denoising function  $f_\theta$ , the denoising function requires to be  $\mathcal{J}$ -invariant function. Then the optimal parameters can be achieved by minimizing the proposed self-supervised loss function. Therefore, we propose a masking scheme to modify a classical repair to  $\mathcal{J}$ -invariant function. Specifically,  $J$  is regarded as a subset sampled from the flawed matrix and the masking scheme to create a blind spot. Then the  $\mathcal{J}$ -invariant function is used to predict the values of the blind spot. Generally, let  $g_\theta$

be the classical repair;  $h(M^f)$  be the function replacing the values on the location  $J$ . Then the  $J$ -invariant function  $f_\theta$  can be defined by

$$f_\theta(M^f)_J := g_\theta(1_J * h(M^f) + 1_{J^c} * M^f)_J \quad (18)$$

where  $1_J$  and  $1_{J^c}$  are the indicator functions of the set  $J$  and  $J^c$ . This means the element-wise multiplication of  $1_J$  with a matrix will be masked and the element-wise multiplication of  $1_{J^c}$  retain element values.

To achieve this, this study proposes a masking scheme named random element selection (RES), which estimates a local distribution  $q(M^f)$  in the neighborhood of an element and then replaces that element with a sample from the distribution. More specifically, we replace the value of element  $l$  with a random element value  $w$  from its  $r * r$  surrounding window. Note that if the value of element  $h$  is also used, information about this element may leak, resulting in a function which is not  $J$ -invariant. Therefore, we replace it with a random neighbor when estimating the local distribution.

The self-supervised loss function can be written as

$$E_{M^f} \| f(RPS_h(M^f))_h - M^f_h \|^2 \quad (19)$$

According to the above stated, minimizing the loss function in Eq. (19) satisfies

$$f^*(M^f)_h = E_{M^f}[M^f_h | RPS_h(M^f)] \quad (20)$$

Let  $q_{lk}(u)$  represents the vector  $u$  with the value  $u_h$  replaced by  $u_k$ . Then

$$f^*(M^f)_h = E_{M^f} E_k[M^f_h | q_{hk}(M^f)] \quad (21)$$

Eq. (3.21) can be converted to

$$\begin{aligned} f^*(M^f)_h &= \frac{1}{r^2} \sum_k E[M^f_h | q_{hk}(M^f)] = \frac{1}{r^2} E[u_h | q_{hh}(M^f)] + \\ &\frac{1}{r^2} \sum_{k \neq h} E[M^f_k | q_{hk}(M^f)] \end{aligned} \quad (22)$$

Since we replace the value  $h$  with its random neighbor, we can regard this value as the element belong to  $-h$  which represents the complement of value  $h$ . Then

$$f^*(M^f)_h = \frac{1}{r^2} E[M^f_h | M^f_{-h}] = (1 - \frac{1}{r^2}) f^*_J(M^f)_h \quad (23)$$

### 3.4 Customized Generative Adversarial Networks

GAN was first proposed by Goodfellow et al. (2014) and was introduced to solve traffic problems, such as traffic state estimation (Liang et al., 2018) and traffic data imputation (Chen et al., 2019; Zhang et al., 2021). Due to the flexible framework,

GAN is capable of incorporating different structures to capture traffic spatio-temporal features. Besides, unlike other neural networks which requires a deterministic input to achieve meaningful outputs, GAN can be trained while interacting only indirectly with the data distribution by sampling from it (Goodfellow et al, 2014). The core idea of GAN is derived from a two-player game model, where one generator  $G$  and one discriminator  $D$  are involved in the game. The generator aims to learn the distribution of real data and then generates fake data with random flawed variables  $z$ . The discriminator attempts to determine whether the input data is real data or generated data. If the input data is real,  $D$  is supposed to classify it to be 1. Otherwise, it should be classified to be 0. To achieve the equilibrium, the two participants require to sequentially learn and optimize themselves to boost their generation capability and discrimination capability, respectively. The performance of  $G$  and  $D$  can be gradually improved by such an adversarial optimization process. Eventually, when  $D$  cannot correctly identify if the input data is generated by  $G$  or is real, we can say that the generator  $G$  has captured the distribution of real data and can be used to estimate. The above adversarial learning process can be formulated as the following min-max function (Vaswani et al., 2017):

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (24)$$

Note that  $G$  and  $D$  can be represented by any differentiable function. Thus, it is a question of whether GAN can capture the implicit distribution of the arterial OD. It has been demonstrated that the optimal discriminator enables the divergence between the distribution of samples from the generator and the distribution of real data to be minimal, as shown in Eq. (25). Thus, the implicit distributions of the arterial OD flow can be captured by GAN.

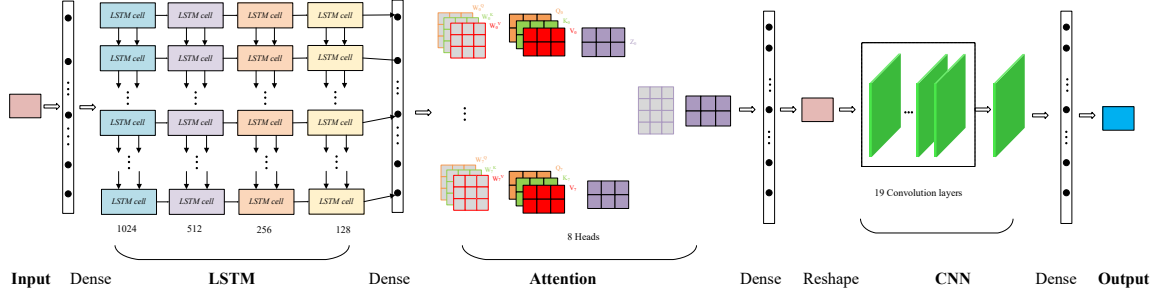
$$\begin{aligned} V(D^*, G) &= -2\log 2 + \int_x p_{data}(x) \log \frac{p_{data}(x)}{(p_{data}(x) + p_G(x))/2} dx \\ &\quad + \int_x p_G(x) \log \frac{p_G(x)}{(p_{data}(x) + p_G(x))/2} \\ &= -2\log 2 + KL(p_{data}(x) || \frac{p_{data}(x) + p_G(x)}{2}) + KL(p_G(x) || \frac{p_{data}(x) + p_G(x)}{2}) \\ &= -2\log 2 + 2JS(p_{data}(x) || p_G(x)) \end{aligned} \quad (3.25)$$

Then, the critical problem is how to design an appropriate structure of the generator and the discriminator. The proposed structures of generator and discriminator are presented in the following subsection.

#### 3.4.1 Structures of the Generator and the Discriminator

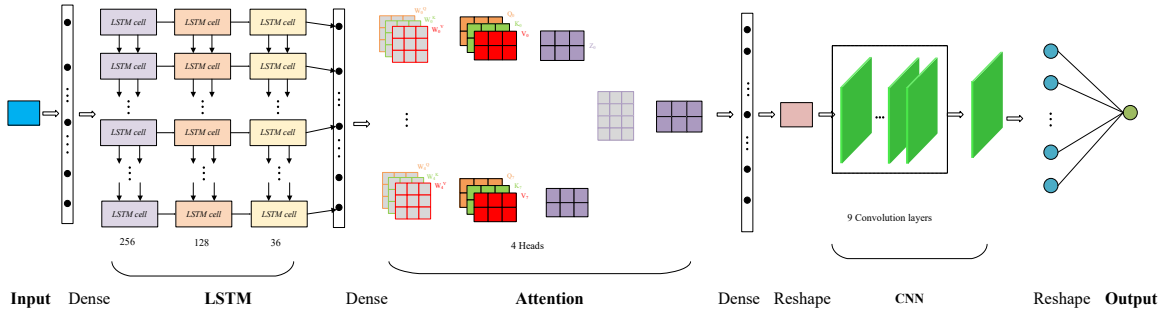
The generator is designed to create realistic arterial OD flow from random repair. ODs are correlated temporally and spatially. More specifically, the arterial OD flow

within a time interval may correlate with the last several time intervals because the path between each OD pair is composed of several links and each link flow comes from the upstream link flows at the last moment and distributes to the downstream link at next moment. Moreover, OD flows also show spatial correlations because paths between all OD pairs along a signalized arterial may intertwine together and interact with each other in a complicated way. The framework of generator is shown in Fig. 6., the input data is processed by the structure of LSTM, attention mechanism, and CNN, the generator can generate outputs.



**Fig. 6. The structure of the customized generator of GAN**

The discriminator is designed to classify the true and the fake generated arterial OD matrix. The generated outputs from generator will be fed into the discriminator for training. The structure of the discriminator is the same as the generator, which means it also consists of LSTM, attention mechanism, and CNN. The difference between them is the output and the inner components of each deep neural network, such as layers and neurons, may vary.



**Fig. 7. The structure of the customized discriminator of GAN**

### 3.4.2 Structures of neural network components

#### *Long short-term memory*

LSTM is frequently applied in time-dependent traffic problems (Hochreiter and Schmidhuber, 1997; Ma et al., 2015). We integrate LSTM in GAN to capture nonlinear temporal correlations to a large extent. LSTM can model highly nonlinear dynamic patterns and encapsulate the information of previous time periods to learn long time dependencies for sequential data. For each LSTM cell, the computation process can be formulated as follows:

$$\tau_i^t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (25)$$

$$\tau_f^t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (26)$$

$$\tau_o^t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (27)$$

$$\tilde{a}_t = \tanh(W_c[\tau_f^t * h_{t-1}, x_t] + b_c) \quad (28)$$

$$h_t = \tau_o^t * (\tau_i^t * \tilde{a}_t + \tau_f^t * a_{t-1}) \quad (29)$$

where  $\tau_i$ ,  $\tau_f$ ,  $\tau_o$  represents the input gate, forget gate, and output gate, respectively;  $\tilde{a}_t$  is the candidate cell state which is used to update the cell state at the current time step;  $h_t$  is the state at current step;  $x_t$  is the input at the current time step;  $W_i$ ,  $W_f$ ,  $W_o$ ,  $W_c$  are weights;  $b_i$ ,  $b_f$ ,  $b_o$ ,  $b_c$  denote biases;  $\sigma$  and  $\tau$  are activation functions.

In this study, we stack multiple layers to capture more temporal features of OD flows from a higher level. The first LSTM layer receives the estimated OD flow according to the collected CV data. Then the following LSTM layers receive the hidden state of the previous layer as their input.

#### *Attention mechanism*

As aforementioned, due to the complex structure of the road network contributes, different OD pairs in the network exert different impacts on the target OD pair, and the weights of impact are highly complicated and dynamic over time. Although this spatial correlation can often be solved through graph convolutional networks by extending conventional convolutional networks to a graph-structured framework, it cannot consider global information to update states and it assigns equal weights to OD pairs. Through the attention mechanism, these problems can be solved. The attention mechanism is motivated by how we pay visual attention when reading a matrix or a sentence. Instead of receiving all information equally, it can assign different weights to each element or word. Thus, attention can improve the efficiency of feature exploration and information processing. Due to this, the attention mechanism has been widely and successfully applied to various domains such as natural language processing. The most popular structure of attention mechanism is developed by Vaswani et al. (2017). In the traffic region, it was mainly used to adaptively explore the most relevant temporal and spatial correlations based on traffic input data and then improve the estimation or prediction accuracy (Jia et al., 2019; Zhang et al., 2020; Zheng et al., 2020; Yin et al., 2021). In this paper, the core idea is to assign different weights to different ODs at each time interval. Up to now, plenty of attention mechanisms have been developed, among which self-attention is adapted most. In this paper, we also borrow the idea of self-attention to explore the spatial correlations between OD pairs. In detail, the attention function softly maps query, keys, and values to outputs, where the query, keys, values, and output are all vectors. For each input vector, three vectors, a query vector, a key vector, and a value vector, are created by multiplying the embedding input vector by three matrices. The three vectors are denoted by  $Q$ ,  $K$ , and  $V$ . Then the score

between the input vectors can be calculated by taking a scaled dot product of the query vector, the key vector, and the value vector.

The flow of all OD pairs has complex and dynamic patterns, which may challenge the expression ability of the self-attention mechanism. To alleviate this, we apply a multi-head attention mechanism to achieve richer representational information. The multi-head attention mechanism obtains several different representations of  $(Q, K, V)$  by mapping  $Q$ ,  $K$ , and  $V$  through several parameter matrices  $W^Q$ ,  $W^K$ ,  $W^V$  computes scaled dot product for each representation, and then concatenates the results.

### *Convolutional neural network*

Although LSTM and attention mechanism have been applied to explore temporal features and spatial features, respectively. We aim to apply CNN to mine the data at the temporal and spatial level deeply and simultaneously to improve the estimation performance. CNN was first proposed in the 1960s. Hubel and Wiesel proposed the concept of “receptive field” (Hubel and Wiesel, 1962). Based on this, Fukushima and Miyake developed the “neocognitron” in the 1980s (Fukushima and Miyake, 1982). Later, Krizhevsky et al. (2012) developed a CNN model to classify images and achieved an outstanding performance. Since then, more and more attention was attracted to CNN. In the transportation region, CNN was applied to predict traffic flow (Duan et al., 2018, Bogaerts et al., 2020), traffic speed (Ma et al., 2017, Cao et al., 2020) In general, the core part of the CNN structure is the convolutional layer and the pooling layer. Unlike traditional multi-layer perception in which the nodes in each layer are fully connected to nodes in the previous layer, convolutional layers are applied to connect neurons to a small region of the previous layer. Typically, multiple convolutional layers are usually applied to connect the input data. One layer can filter one feature. Therefore, multiple layers will filter multiple features. Thus, the high performance of CNN can be achieved. Such a process can be formulated as:

$$y_{conv} = \sigma(W_c * X + b) \quad (30)$$

where  $y_{conv}$  denotes the output of convolutional layer;  $W_c$  denotes the weights;  $X$  denotes the input data which is the output of the attention mechanism layer;  $b$  denotes the bias.

Pooling layers are used to downsampling the convolution layers and decrease the dimensionality of each feature map to achieve spatial and scale invariance, lower computation, and control overfitting. In this paper, the pooling layer is not applied since the dimension of the input data is not large. Finally, the extracted traffic features through convolutional and pooling layers are concatenated into a dense vector and then transformed into model outputs through a fully connected layer.

### 3.5 Simple toy example

To better understand the estimation process, we use a simple example to explain. There is an arterial with 6 OD pairs. We assume that human-driven vehicles and CVs

travel along this arterial. Roadside units (RSU) are installed to collect the information CV broadcasts. And RSU constantly transfers the information to the central computer. At the end of the time period, the central computer will calculate the CV OD flow for each time interval during the time period. We assume a time of 30 minutes with intervals of 5 minutes and a CV market penetration rate of 20%. The collected CV OD flow is shown in Table 2.

Table 2 Collected CV OD flow

|   | 0 – 5 min | 5- 10 min | 10- 15 min | 15 – 20min | 20- 25 min | 25- 30 min |
|---|-----------|-----------|------------|------------|------------|------------|
| 1 | 13        | 20        | 16         | 15         | 25         | 22         |
| 2 | 12        | 22        | 20         | 18         | 22         | 20         |
| 3 | 14        | 22        | 16         | 18         | 26         | 26         |
| 4 | 12        | 24        | 18         | 20         | 22         | 24         |
| 5 | 10        | 26        | 22         | 22         | 24         | 22         |
| 6 | 12        | 24        | 22         | 24         | 30         | 26         |

Since the CV market penetration rate is 20%, the roughly estimated OD flow (i.e., flawed OD matrix) can be calculated as shown in Table X.

Table 3 Roughly estimated OD flow based on collected CV OD flow

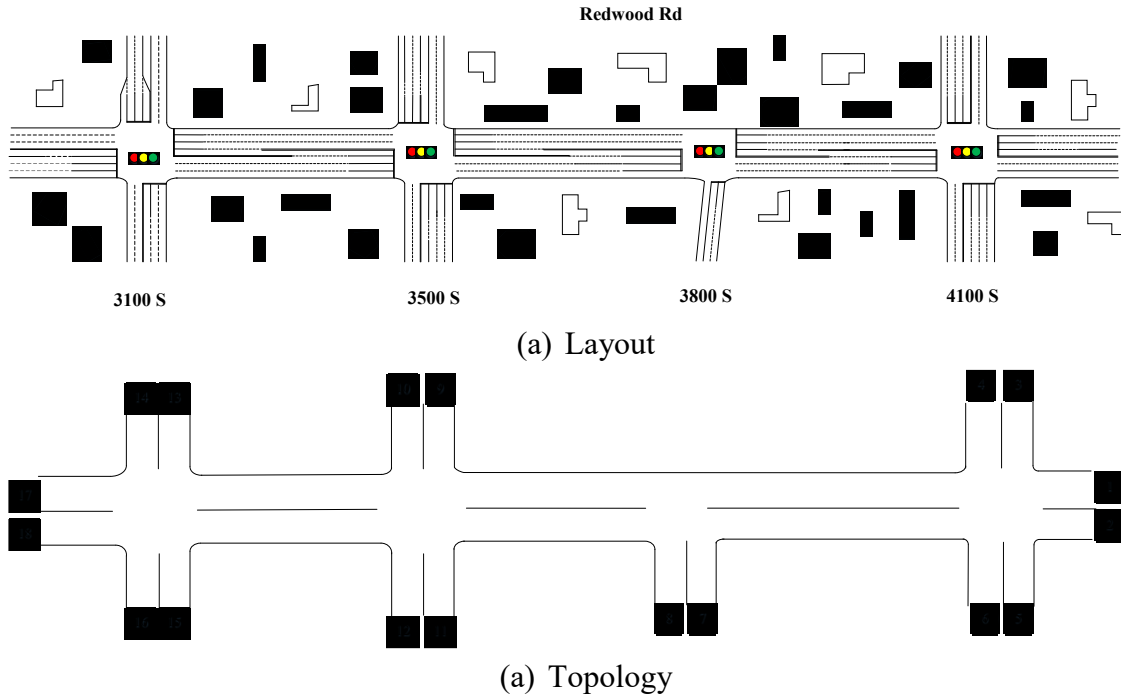
|   | 0 – 5 min | 5- 10 min | 10- 15 min | 15 – 20min | 20- 25 min | 25- 30 min |
|---|-----------|-----------|------------|------------|------------|------------|
| 1 | 65        | 100       | 80         | 75         | 125        | 110        |
| 2 | 60        | 110       | 100        | 90         | 110        | 100        |
| 3 | 70        | 110       | 80         | 90         | 130        | 130        |
| 4 | 60        | 120       | 90         | 100        | 110        | 120        |
| 5 | 50        | 130       | 110        | 110        | 120        | 110        |
| 6 | 60        | 120       | 110        | 120        | 150        | 130        |

Then this flawed OD matrix is then fed into the developed GAN network. Through LSTM, pay attention to the network and CNN, generator will produce an output that will be fed into the discriminator. Based on the proposed loss function and GAN loss, the generator and discriminator interact to improve their performance. Finally, the performance of generator will be greatly improved, and then output the estimated OD flow.

## 4 Numerical Example

### 4.1 Experimental Design

To evaluate the effectiveness and to assess the potential for field applications of the proposed model, this study selects a segment on Redwood Road in Salt Lake City, Utah for study. Fig. 8 shows its geometric layout and its topology. This arterial consists of four intersections, 72 OD pairs. Due to the difficulties in collecting ground-truth OD flows from the real world, we used the simulation platform, VISSIM, to replicate the field traffic conditions. Since the simulation system is useful and meaningful only if it can faithfully reflect the realistic driving environment, the simulator is calibrated with field data. The field data is collected from the Automated Traffic Signal Performance Measures (ATSPM) system which uses Wavetrnix SmaterSensor Matrix detectors located at traffic signal stop-bar to collect and restore lane-by-lane turning counts and uses Wavetrnix SmartSensor Advance detectors located approximately 300 ft upstream of signalized intersections to provide total through traffic counts.



**Fig. 8. The geometric layout of the studied site**

Based on the calibrated simulation arterial, simulation can be used to generate simulated scenarios for model evaluation. In the simulation, we define two types of vehicles, CV and regular vehicles. The trajectory of each CV can be collected through running the simulation and those CV trajectories can directly yield the CV OD flow. We use 160 different seeds to run simulations to collect data. The simulation period for each seed is set as 3 hours. Then we further divide it into 18 identical intervals (each interval is 10 minutes) to achieve CV OD flow based on trajectory data. The minimum time frame for collecting CV data depends on the arterial scale. To ensure the accuracy of the

proposed model, it is required that most paths along the arterial are traveled by at least one CV. That means for most OD pairs, the OD flow should be more than 0. To obtain more data for training, we set the time period as three hours, while one-hour time frame or less also works well technically.

Considering the only available information is CV trajectory data, we conduct comparisons between the following models to evaluate the performance of the proposed model. Cases 2-5 are compared to fine-tune the structure of GAN.

- (1) Simple scaled model (SSM): The OD flow is estimated by scaling up the CV OD flow with the corresponding global CV market penetration rate.
- (2) GAN framework incorporating LSTM (GL): The generator and discriminator of the GAN network is the LSTM network.
- (3) GAN framework incorporating attention mechanism (GA): The generator and discriminator of the GAN network is the attention network.
- (4) GAN framework incorporating CNN (GC): The generator and discriminator of the GAN network is the CNN network.
- (5) GAN framework incorporating LSTM, attention mechanism, and CNN (GLAC): The generator and discriminator of the GAN network embed with LSTM, attention, and CNN.

Three error indicators are used to measure the loss: mean absolute (MAE), mean absolute percentage error (MAPE), and mean square error (MSE). The smaller value of the metric means the better accuracy. The definitions of those performance indicators are summarized in the following equations.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (31)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (32)$$

$$MAPE = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (33)$$

where  $y_i$  and  $\hat{y}_i$  denote the ground-truth value and estimated value, respectively;  $N$  denotes the number of samples.

## 4.2 Result Analysis

The training process of the GLAC model, with 500 iterations, costs 7780 seconds on a workstation equipped with NVIDIA GeForce RTX 2080 Ti GPU. In the testing phase, the estimation time of the model is about 0.8 seconds. Note that the computational complexity of the model is largely influenced by deep neural network structures, such as the number of LSTM layers, attention heads, and convolutional layers. The influence of the number of nodes on the computational complexity is minimal as the number of nodes along the arterial only affects the size of the network input. Our developed GAN network includes LSTM, Attention mechanism, and CNN. The input size has much less influence on the computational complexity of the LSTM network and attention mechanism due to the fixed number of layers, neurons in each

layer, and heads. Before feeding into the CNN, the output of the attention mechanism will be reshaped into a matrix that has the same dimension as the input. Therefore, as the number of nodes increases, the computational complexity increases. This is because more nodes mean more paths along this arterial, leading to a larger input size. When we keep other hyperparameters as the same, such as kernel size, padding, stride, etc., a larger input size means more operations including multiplications and summations. The market penetration rate only affects the value of input data and therefore has no impact on the computational complexity.

Table 4 summarizes the performance of those models in the same dataset when the CV market penetration rate is 25%. It can be observed that the GLAC model performs the best. It can yield 13.25 veh/10min of MSE, 33.69% of MAPE, and 2.73 veh/10min of MAE for OD. Compared to the SSM, this model could bring a 45.51%, 16.22%, and 29.27% improvement in those three metrics, respectively. This model also outperforms GL, GA, and GC in terms of those metrics. This demonstrates that proposed structure of GLAC can capture richer temporal and spatial information. Moreover, GL, GA, and GC indicate some improvements in those performance indicators compared with SSM.

**Table 4 Performance comparison of various models**

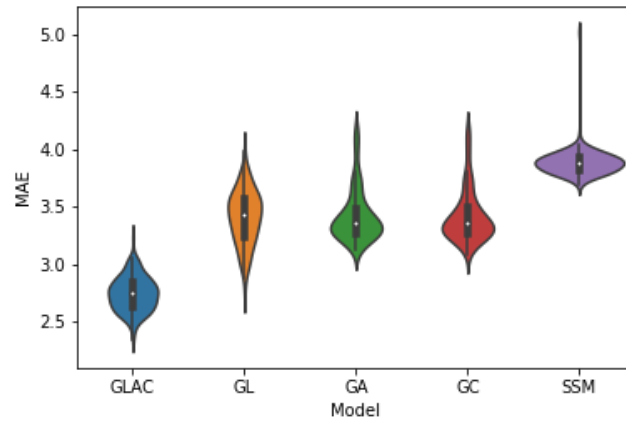
| Model | MAE         | MSE          | MAPE          |
|-------|-------------|--------------|---------------|
| SSM   | 3.86        | 24.32        | 49.91%        |
| GL    | 3.39        | 20.54        | 38.08%        |
| GA    | 3.4         | 19.46        | 40.64%        |
| GC    | 3.4         | 19.42        | 40.61%        |
| GLAC  | <b>2.73</b> | <b>13.25</b> | <b>33.69%</b> |

An extended comparison of those five models is shown in Table 5 and Fig. 9. Table 3 summarizes the aggregate performance of the five models for all test OD pairs. Comparison of the three measures regarding all statistical indicators indicates that the GLAC model outperforms all other models. Fig. 9 visualizes the distribution of the five measures of all test OD pairs and reiterates that the proposed GLAC model shows the best performance.

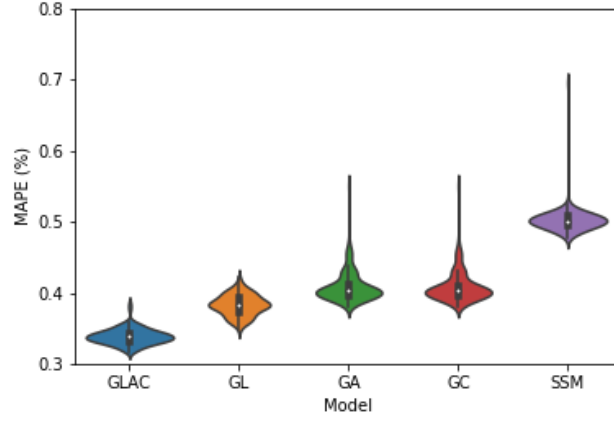
**Table 5 The overall performance of the five models**

| Model | Statistical indicator    | MAE  | MSE   | MAPE   |
|-------|--------------------------|------|-------|--------|
| SSM   | Minimum                  | 3.69 | 22.22 | 47.63% |
|       | 25th percentile          | 3.82 | 23.75 | 49.49% |
|       | 50th percentile (Median) | 3.88 | 24.36 | 50.10% |
|       | 75th percentile          | 3.94 | 25.08 | 50.81% |
|       | Maximum                  | 5.02 | 39.97 | 69.53% |
| GL    | Minimum                  | 2.75 | 13.63 | 34.76% |

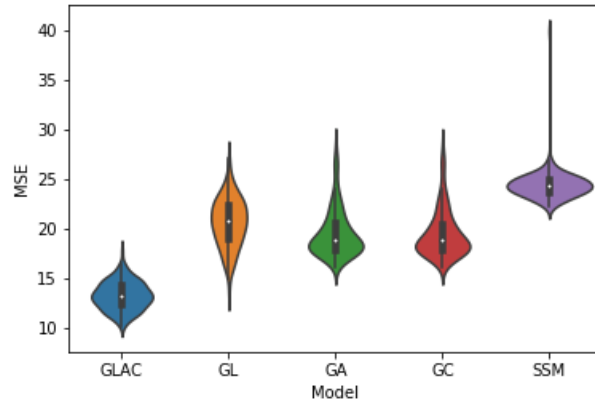
|      |                                         |             |              |               |
|------|-----------------------------------------|-------------|--------------|---------------|
|      | 25th percentile                         | 3.25        | 19.03        | 37.26%        |
|      | 50th percentile<br>(Median)             | 3.43        | 20.81        | 38.32%        |
|      | 75th percentile                         | 3.57        | 22.44        | 39.32%        |
|      | Maximum                                 | 3.99        | 27.05        | 42.15%        |
| GA   | Minimum                                 | 3.12        | 16.17        | 38.14%        |
|      | 25 <sup>th</sup> percentile             | 3.28        | 18           | 39.65%        |
|      | 50 <sup>th</sup> percentile<br>(Median) | 3.36        | 18.85        | 40.28%        |
|      | 75th percentile                         | 3.50        | 20.61        | 41.33%        |
|      | Maximum                                 | 4.17        | 28.41        | 54.99%        |
| GC   | Minimum                                 | 3.09        | 16.17        | 38.14%        |
|      | 25 <sup>th</sup> percentile             | 3.28        | 17.93        | 39.65%        |
|      | 50 <sup>th</sup> percentile<br>(Median) | 3.36        | 18.85        | 40.32%        |
|      | 75th percentile                         | 3.49        | 20.56        | 51.52%        |
|      | Maximum                                 | 4.16        | 28.35        | 55.02%        |
| GLAC | Minimum                                 | <b>2.35</b> | <b>10.22</b> | <b>31.53%</b> |
|      | 25 <sup>th</sup> percentile             | <b>2.64</b> | <b>12.40</b> | <b>33.25%</b> |
|      | 50 <sup>th</sup> percentile<br>(Median) | <b>2.75</b> | <b>13.24</b> | <b>33.82%</b> |
|      | 75th percentile                         | <b>2.85</b> | <b>14.43</b> | <b>34.45%</b> |
|      | Maximum                                 | <b>3.23</b> | <b>17.80</b> | <b>38.56%</b> |



(a) MAE



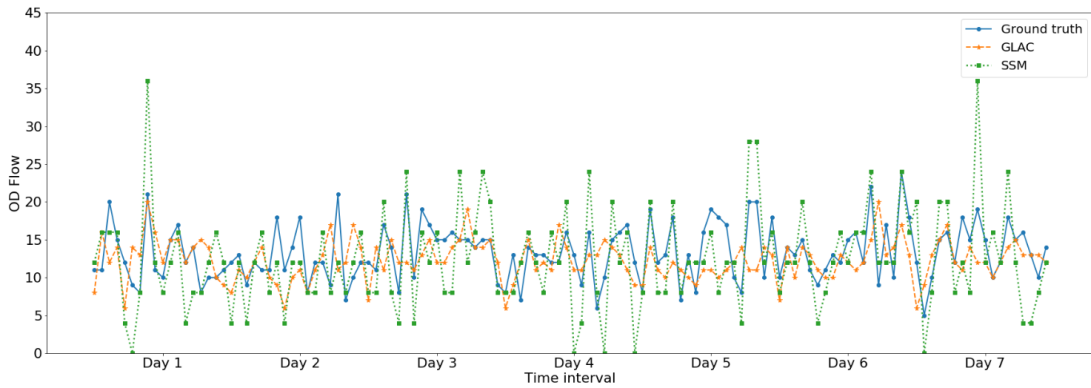
(b) MAPE



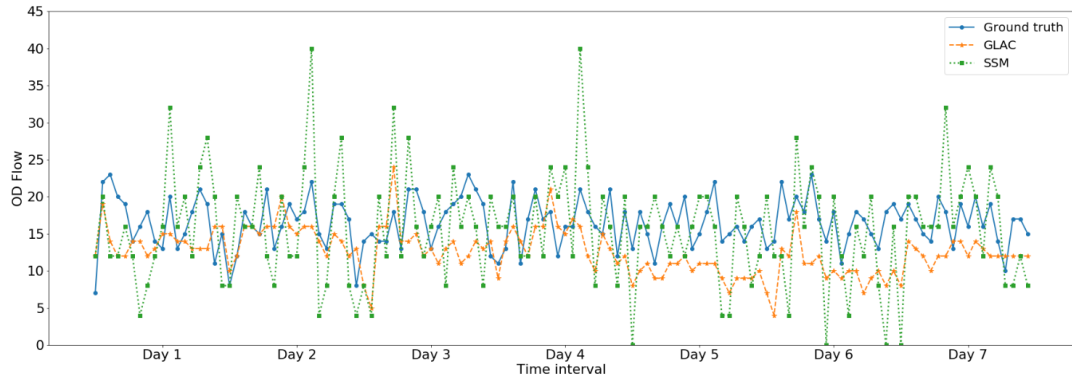
(c) MSE

**Fig. 9. Violin plots comparing the overall performance of the five models**

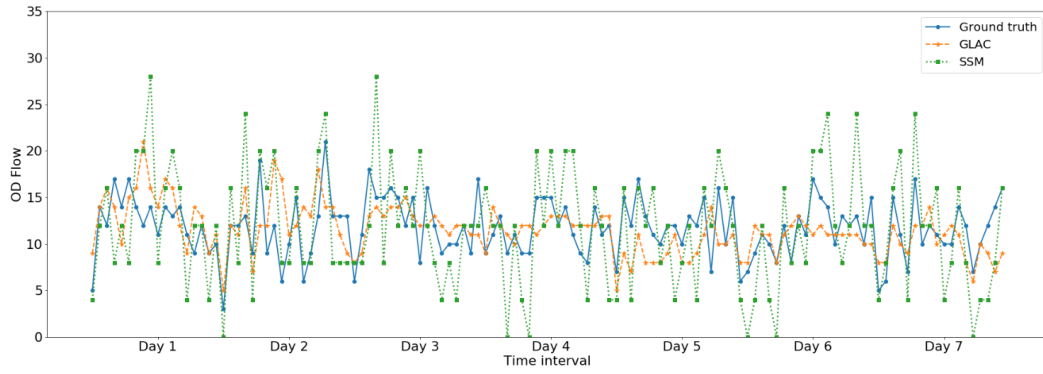
To evaluate the performance of the proposed GLAC model better and further in estimating OD flows, we select a set of OD pairs with the best 3 and worst 3 performance (in MAPE sense) to compare the estimated and actual OD flow (samples of one week), as shown in Fig. 9. Among them, Fig 10 (a), (b), and (c) are time-dependent OD flows with the best performance. Fig 10 (d), (e), (f) are those with worst performance. All those figures indicate that the proposed GLAC model can yield sufficient reliable estimates. Although for the condition with the worst performance, this model still produces acceptable estimates compared with the scaled model.



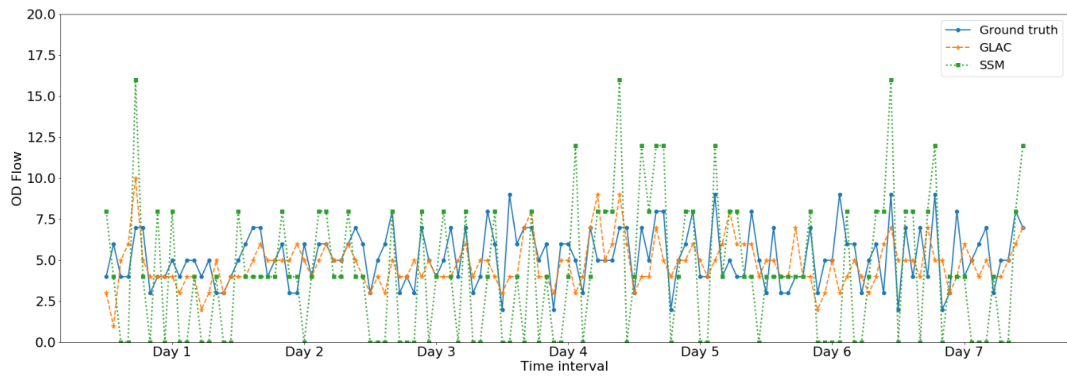
(a) Time-dependent flows for OD pair 1-6 (**best**)



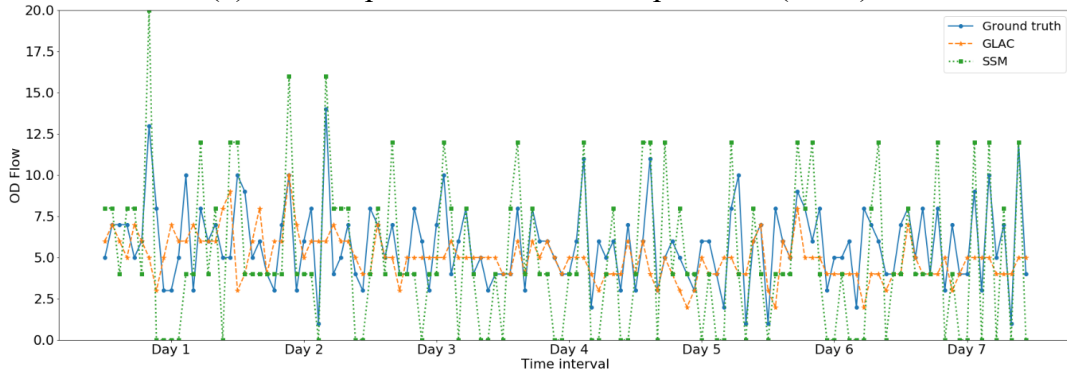
(b) Time-dependent flows for OD pair 18-11(**best**)



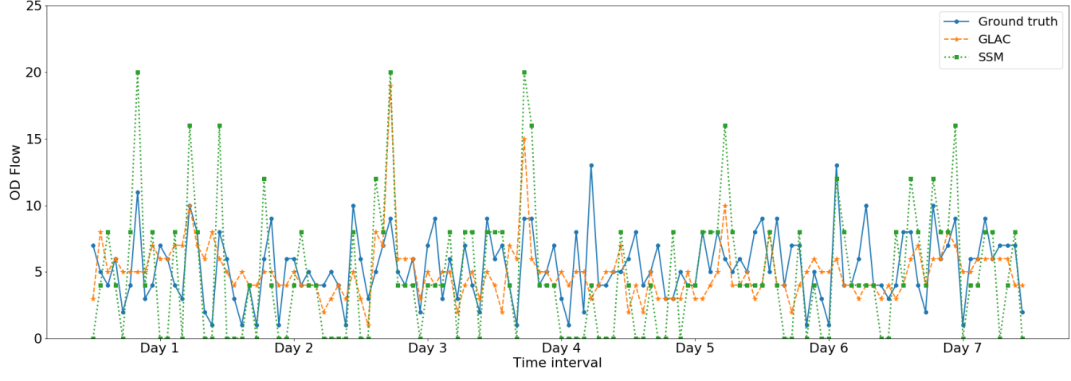
(c) Time-dependent flows for OD pair 1-6 (**best**)



(d) Time-dependent flows for OD pair 15-2 (**worst**)



(e) Time-dependent flows for OD pair 15-17 (**worst**)

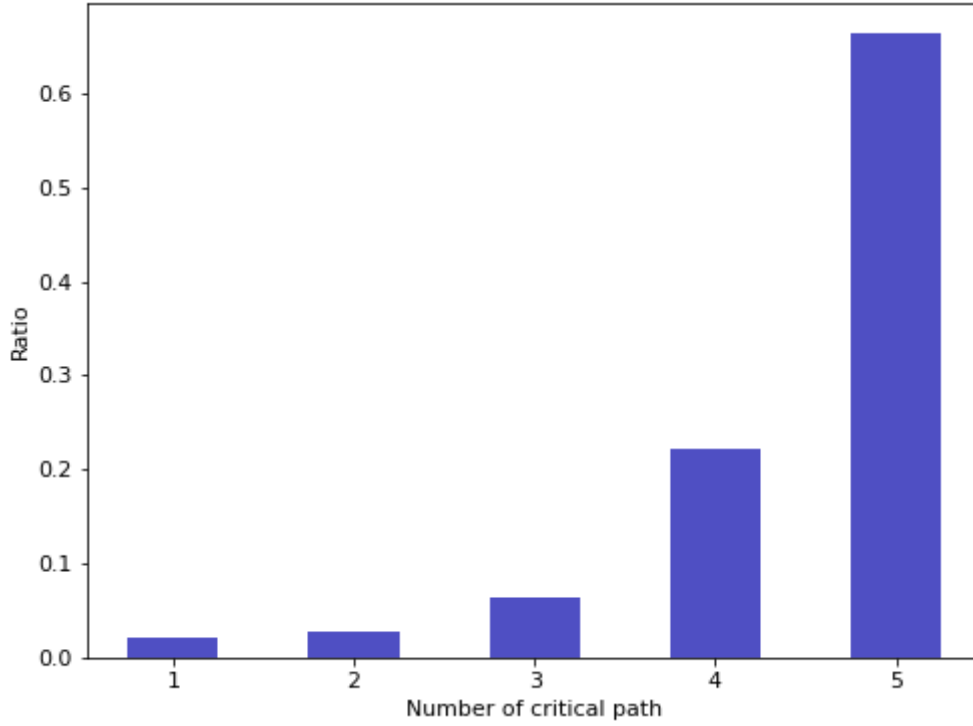


(f) Time-dependent flows for OD pair 15-13 (worst)

**Fig. 10. Time-dependent flows for OD pairs with the best and worst performance**

In the design of a multi-path progression traffic signal system, it is essential to determine the path-flow patterns. Therefore, we then check the reliability of the proposed model for identifying path-flow patterns. For each time interval, we firstly sort those paths in the descending order based on the estimated volumes and we can obtain an initial path sequence. The path sequence will later be updated based on the actual volumes. Then, we can access how many critical paths can be identified accurately by the proposed model and count the number of time intervals the proposed model performs successfully. An accuracy ratio can be calculated by dividing the number of successes and the total number of time interval.

Fig. 11 shows the ratio of the proposed model for determining the five critical paths within each time interval during those days. It can be observed that the ratio that the proposed model identifies all five critical paths is 66.32%, and the ratio that it identifies three and four of the five actual critical paths are 22.24% and 6.38%, respectively. The number of critical paths is determined by our developed model according to the geometry of the arterial. The model shows that when the number of critical paths is 5, this arterial can have the best coordination performance. Therefore, this indicates that the proposed model could produce a satisfying estimation of path-flow patterns with correct rankings.



**Fig. 11. Ratio of identified critical paths from the proposed model**

#### 4.3 Sensitivity Analysis

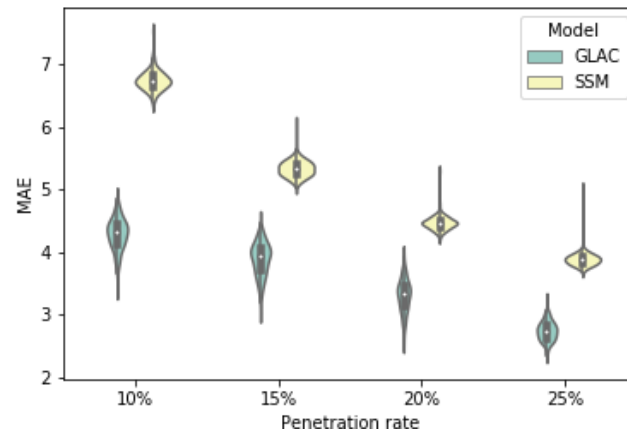
In this section, we conduct sensitivity analysis to evaluate the effectiveness of the proposed model in estimating OD flows with various CV market penetration rates, which range from 10% to 25%. The SSM is conducted using those market penetration rates for comparison. The results are shown in Table 6. As the CV market penetration rate rises, all three performance indicators for both the scaled model and GLAC improve. This is because a higher CV market penetration rate will result in input data with less flaw. Table 4 shows that when the CV penetration rate is 15%, the MAPE of GLAC can be about 40%, which indicates reliable estimation results.

**Table 6 Sensitivity analysis on various market penetration rates**

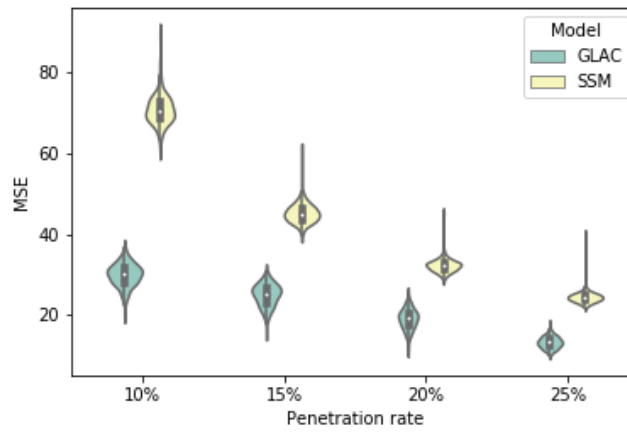
| Metric | Model | Market penetration rate |               |               |               |
|--------|-------|-------------------------|---------------|---------------|---------------|
|        |       | 10%                     | 15%           | 20%           | 25%           |
| MAPE   | SSM   | 86.44%                  | 67.54%        | 57.26%        | 49.91%        |
|        | GLAC  | <b>45.30%</b>           | <b>41.51%</b> | <b>36.73%</b> | <b>33.69%</b> |
| MSE    | SSM   | 70.55                   | 45            | 32.13         | 24.32         |
|        | GLAC  | <b>29.83</b>            | <b>24.87</b>  | <b>19</b>     | <b>13.25</b>  |
| MAE    | SSM   | 6.7                     | 5.3           | 4.44          | 3.86          |
|        | GLAC  | <b>4.26</b>             | <b>3.97</b>   | <b>3.29</b>   | <b>2.73</b>   |

An extended comparison of the two models under different market penetration rates is shown via violin plots (Fig. 12), which visualize the distribution of the three measures over all test OD pairs and indicate that the GLAC model shows superior

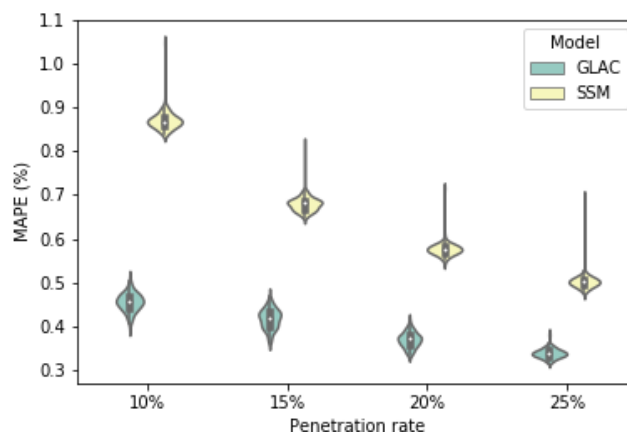
performance.



(a) MAE



(b) MSE



(c) MAPE

**Fig. 12. The performance metric distributions with various market penetration rates**

## 5 Conclusion

The real-time arterial OD flow is critical for traffic control, but the ground-truth is difficult to observe. To address this issue, we treated the OD flow simply estimated from the CV OD as a flawed matrix, and applied a masking scheme to process the flawed matrix to create a self-learning method incorporating the  $J - invariant$  function. To capture the spatio-temporal correlation of the OD flows, we customized a nonparametric model based on GAN to estimate the time-dependent OD flows along a signalized arterial, and trained the customized GAN with the proposed self-supervised learning method. The proposed GAN consists of three sub-networks: LSTM to learn the evolution pattern from a temporal perspective, attention mechanism to capture spatial features, and CNN to bridge both temporal and spatial features. During the training process, we only used information from collected CV trajectory as training data.

To evaluate the effectiveness of the proposed model and the self-supervised learning method, a real-world signalized arterial is selected, set up, and calibrated in VISSIM, based on an arterial in Salt Lake City, Utah, for generating CV trajectory. Experiments indicate promising results. When the CV market penetration rate is low to 25%, the proposed model GLAC model can yield 13.25 veh/10min of RMSE, 33.69% of MAPE, and 2.73 veh/10min of MAE for OD. Compared to the SSM, this model could bring a 45.51%, 16.22%, and 29.27% improvement in MAE, RMSE, MAPE, respectively. The results also demonstrated that our proposed model could produce a satisfying estimation of path-flow patterns with correct rankings. More specifically, the ratio that the proposed model identifies all five critical paths is 66.32%, and the ratios that it identifies three and four of the five actual critical paths are 22.24% and 6.38%, respectively. The numerical study implies a reliable estimation accuracy for arterial OD flow. This could provide more efficient arterial management, such as multi-path signal coordination which could improve the vehicle operation efficiency when traveling along the arterial.

Moreover, sensitivity analysis also demonstrates the promising performance is achieved by the proposed model with various CV market penetration rates. The sensitivity analysis implies that with the market penetration rates increases, the performance of the proposed model improves. In addition, to obtain an acceptable estimation accuracy by applying this method. The minimum CV market penetration rate should be about 15%.

In this study, the proposed model was only applied to signalized arterials. Future work could focus on validating the proposed method with field data on the urban road network. Because traffic conditions and patterns in the urban road network are more complex. For example, there is only one path between each OD pair along the arterial, we do not need to make any hypothesis on users' path choice behavior. In the network-wide OD flow estimation, there are several paths for travelers to choose between each OD and it is needed to capture travelers' path choice behaviors. Future research should be conducted to further test the effectiveness and efficiency of the proposed model and method. Using CV data only to estimate network OD flow will be beneficial for the traffic management, such as route guidance, congestion management.

## **Authorship contribution statement**

Qinzheng Wang: Methodology, Investigation, Writing- Original draft preparation. Yun Yuan: Data curation, Visualization, Validation. Qiwei Zhang: Validation, Writing-Reviewing. Xianfeng Terry Yang: Conceptualization, Supervision, Methodology, Writing- Reviewing and Editing.

## **Acknowledgement**

This research is supported by the research grant “CMMI #2234289: CAREER: Physics Regularized Machine Learning Theory: Modeling Stochastic Traffic Flow Patterns for Smart Mobility Systems” which is sponsored by the National Science Foundation.

## Reference

- Alibabai, H., Mahmassani, H.S., 2008. Dynamic origin-destination demand estimation using turning movement counts. *Transportation research record* 2085, 39-48.
- Antoniou, C., Ben-Akiva, M., Koutsopoulos, H.N., 2004. Incorporating automated vehicle identification data into origin-destination estimation. *Transportation Research Record* 1882, 37-44.
- Arsava, T., Xie, Y., Gartner, N.H., 2016. Arterial progression optimization using OD-BAND: case study and extensions. *Transportation Research Record*, 2558(1), pp.1-10.
- Asakura, Y., Hato, E., Kashiwadani, M., 2000. Origin-destination matrices estimation model using automatic vehicle identification data and its application to the han-shin expressway network. *Transportation* 27, 419-438.
- Ashok, K., Ben-Akiva, M.E., 2002. Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping to path flows and link flows. *Transportation science* 36, 184-198.
- Asmundsdottir, R., Chen, Y., van Zuylen, H.J., 2010. Dynamic origin-destination matrix estimation using probe vehicle data as a priori information, in: *Traffic data collection and its standardization*. Springer, pp. 89-108.
- Batson, J., Royer, L., 2019, May. Noise2self: Blind denoising by self-supervision. In *International Conference on Machine Learning* (pp. 524-533). PMLR.
- Baek, S., Lim, Y., Rhee, S., Choi, K., 2010. Method for estimating population od matrix based on probe vehicles. *KSCE Journal of Civil Engineering* 14, 231-235.
- Bell, M.G., 1983. The estimation of an origin-destination matrix from traffic counts. *Transportation Science* 17, 198-217.
- Bell, M.G., 1991. The real time estimation of origin-destination flows in the presence of platoon dispersion. *Transportation Research Part B: Methodological* 25, 115-125.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. New York, NY: Springer.
- Bogaerts, T., Masegosa, A.D., Angarita-Zapata, J.S., Onieva, E. and Hellinckx, P., 2020. A graph CNN-LSTM neural network for short and long-term traffic forecasting based on trajectory data. *Transportation Research Part C: Emerging Technologies*, 112, pp.62-77.
- Cao, M., Li, V.O. and Chan, V.W., 2020, May. A CNN-LSTM model for traffic speed prediction. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)* (pp. 1-5). IEEE.
- Cao, P., Miwa, T., Yamamoto, T., Morikawa, T., 2013. Bilevel generalized least squares estimation of dynamic origin-destination matrix for urban network with probe vehicle data. *Transportation research record* 2333, 66-73.
- Cascetta, E., 1984. Estimation of trip matrices from traffic counts and survey data: a generalized least squares estimator. *Transportation Research Part B: Methodological* 18, 289-299.
- Cascetta, E., Inaudi, D., Marquis, G., 1993. Dynamic estimators of origin-destination

- matrices using traffic counts. *Transportation science* 27, 363-373.
- Cascetta, E., Nguyen, S., 1988. A unified framework for estimating or updating origin/destination matrices from traffic counts. *Transportation Research Part B: Methodological* 22, 437-455.
- Cascetta, E., Papola, A., Marzano, V., Simonelli, F., Vitiello, I., 2013. Quasi-dynamic estimation of od flows from traffic counts: Formulation, statistical validation and performance analysis on real data. *Transportation Research Part B: Methodological* 55, 171-187.
- Castillo, E., Jimenez, P., Menendez, J.M., Nogal, M., 2013. A bayesian method for estimating traffic flows based on plate scanning. *Transportation* 40, 173-201.
- Castillo, E., Menendez, J.M., Sanchez-Cambronero, S., 2008b. Traffic estimation and optimal counting location without path enumeration using Bayesian networks. *Computer-Aided Civil and Infrastructure Engineering* 23, 189-207.
- Castillo, E., Menendez, J.M., Sanchez-Cambronero, S., Calvino, A., Sarabia, J.M., 2014. A hierarchical optimization problem: estimating traffic flow using gamma random variables in a Bayesian context. *Computers & Operations Research* 41, 240-251.
- Chang, G.L., Wu, J., 1994. Recursive estimation of time-varying origin-destination flows from traffic counts in freeway corridors. *Transportation Research Part B: Methodological* 28, 141-160.
- Chen, Y., Lv, Y., Wang, F.Y., 2019. Traffic flow imputation using parallel data and generative adversarial networks. *IEEE Transactions on Intelligent Transportation Systems*, 21(4), pp.1624-1630.
- Chiou, Y.C., Lan, L.W., Tseng, C.M., Fan, C.C., 2011. Optimal locations of license plate recognition to enhance the origin-destination matrix estimation, in: *Proceedings of the Eastern Asia Society for Transportation Studies Vol. 8 (The 9th International Conference of Eastern Asia Society for Transportation Studies, 2011)*, Eastern Asia Society for Transportation Studies. pp. 297-297.
- Dixon, M.P., Rilett, L.R., 2002. Real-time od estimation using automatic vehicle identification and traffic count data. *Computer-Aided Civil and Infrastructure Engineering* 17, 7-21. 31.
- Duan, Z., Yang, Y., Zhang, K., Ni, Y., Bajgain, S., 2018. Improved deep hybrid networks for urban traffic flow prediction using trajectory data. *Ieee Access*, 6, pp.31820-31827.
- Feng, Y., Head, K.L., Khoshmashgham, S., Zamanipour, M., 2015. A real-time adaptive signal control in a connected vehicle environment. *Transportation Research Part C: Emerging Technologies*, 55, pp.460-473.
- Frederix, R., Viti, F., Tampère, C.M., 2013. Dynamic origin-destination estimation in congested networks: theoretical findings and implications in practice. *Transportmetrica A: Transport Science*, 9(6), pp.494-513.
- Fukushima, K., Miyake, S., 1982. Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern recognition*, 15(6), pp.455-469.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S.,

- Courville, A., Bengio, Y., 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Gong, Z., 1998. Estimating the urban OD matrix: A neural network approach. *European Journal of operational research*, 106(1), pp.108-115.
- Hazelton, M.L., 2008. Statistical inference for time varying origin-destination matrices. *Transportation Research Part B: Methodological* 42, 542-552.
- Hendriksen, A.A., Pelt, D.M., Batenburg, K.J., 2020. Noise2inverse: Self-supervised deep convolutional denoising for tomography. *IEEE Transactions on Computational Imaging*, 6, pp.1320-1335.
- Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural computation*, 9(8), pp.1735-1780.
- Huang, T., Ma, Y., Qin, Z.T., Zheng, J., Liu, H.X., Zhu, H., Ye, J., 2019, December. Origin-destination Flow Prediction with Vehicle Trajectory Data and Semi-supervised Recurrent Neural Network. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1450-1459). IEEE.
- Hubel, D.H., Wiesel, T.N., 1962. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1), pp.106-154.
- Iqbal, M.S., Choudhury, C.F., Wang, P., Gonzalez, M.C., 2014. Development of origin-destination matrices using mobile phone call data. *Transportation Research Part C: Emerging Technologies* 40, 63-74.
- Jia, X., Li, T., Zhu, R., Wang, Z., Zhang, Z., Wang, J., 2019, December. Traffic Flow Prediction Based on Self-attention Mechanism and Deep Packet Residual Network. In *Proceedings of the 2019 7th International Conference on Information Technology: IoT and Smart City* (pp. 575-580).
- Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, pp.1097-1105.
- Krull, A., Buchholz, T.O., Jug, F., 2019. Noise2void-learning denoising from single noisy images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2129-2137).
- Kullback, S., Leibler, R.A., 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1), pp.79-86.
- Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M. and Aila, T., 2018. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*.
- Liang, Y., Cui, Z., Tian, Y., Chen, H., Wang, Y., 2018. A deep generative adversarial architecture for network-wide spatial-temporal traffic-state estimation. *Transportation Research Record*, 2672(45), pp.87-105.
- Lin, P.W., 2006. A robust model for estimating freeway dynamic origin-destination matrices. Ph.D. thesis.
- Lou, Y., Yin, Y., 2010. A decomposition scheme for estimating dynamic origin-destination flows on actuation-controlled signalized arterials. *Transportation Research Part C: Emerging Technologies*, 18(5), pp.643-655.

- Lu, Z., Rao, W., Wu, Y.J., Guo, L., Xia, J., 2015. A kalman filter approach to dynamic od ow estimation for urban road networks using multi-sensor data. *Journal of Advanced Transportation* 49, 210-227.
- Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., Wang, Y., 2017. Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, 17(4), p.818.
- Ma, X., Tao, Z., Wang, Y., Yu, H., Wang, Y., 2015. Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C: Emerging Technologies*, 54, pp.187-197.
- Maher, M.J., 1983. Inferences on trip matrices from observations on link volumes: a Bayesian statistical approach. *Transportation Research Part B: Methodological* 17, 435-447.
- Metzler, C.A., Mousavi, A., Heckel, R. and Baraniuk, R.G., 2018. Unsupervised learning with Stein's unbiased risk estimator. *arXiv preprint arXiv:1805.10531*.
- Okutani, I., Stephanedes, Y.J., 1984. Dynamic prediction of traffic volume through kalman filtering theory. *Transportation Research Part B: Methodological* 18, 1-11.
- Ou, J., Lu, J., Xia, J., An, C., Lu, Z., 2019. Learn, assign, and search: real-time estimation of dynamic origin-destination flows using machine learning algorithms. *IEEE Access*, 7, pp.26967-26983.
- Padinjarapat, R. K., & Mathew, S. (2013). OD matrix estimation from link counts using artificial neural network. *International Journal of Scientific and Engineering*, 4(5), 293–296.
- Park, E.S., Rilett, L.R., Spiegelman, C.H., 2008. A markov chain monte carlo-based origin destination matrix estimator that is robust to imperfect intelligent transportation systems data. *Journal of Intelligent Transportation Systems* 12, 139-155.
- Rao, W., Wu, Y.J., Xia, J., Ou, J., Kluger, R., 2018. Origin-destination pattern estimation based on trajectory reconstruction using automatic license plate recognition data. *Transportation Research Part C: Emerging Technologies* 95, 29-46.
- Sherali, H.D., Park, T., 2001. Estimation of dynamic origin{destination trip tables for a general network. *Transportation Research Part B: Methodological* 35, 217-235.
- Sohn, K., Kim, D., 2008. Dynamic origin{destination ow estimation using cellular communication system. *IEEE Transactions on Vehicular Technology* 57, 2703-2713.
- Soltanayev, S. and Chun, S.Y., 2018. Training deep learning based denoisers without ground truth data. *arXiv preprint arXiv:1803.01314*.
- Spiess, H., 1987. A maximum likelihood model for estimating origin-destination matrices. *Transportation Research Part B: Methodological* 21, 395-412. 32
- Sun, S., Yu, G., Zhang, C., 2004. Short-term traffic ow forecasting using sampling markov chain method with incomplete data, in: *IEEE Intelligent Vehicles Symposium*, 2004, IEEE. pp. 437-441.

- Tang, K., Cao, Y., Chen, C., Yao, J., Tan, C., Sun, J., 2021. Dynamic origin-destination flow estimation using automatic vehicle identification data: A 3D convolutional neural network approach. *Computer-Aided Civil and Infrastructure Engineering*, 36(1), pp.30-46.
- Van Der Zijpp, N.J., 1997. Dynamic origin-destination matrix estimation from traffic counts and automated vehicle identification data. *Transportation Research Record* 1607, 87-94.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. In *Advances in neural information processing systems* (pp. 5998-6008).
- Wang, Q., Yuan, Y., Yang, X.T., Huang, Z., 2021. Adaptive and multi-path progression signal control under connected vehicle environment. *Transportation Research Part C: Emerging Technologies*, 124, p.102965.
- Xie, Y., Wang, Z., Ji, S., 2020. Noise2same: Optimizing a self-supervised bound for image denoising. *arXiv preprint arXiv:2010.11971*.
- Yang, H., Akiyama, T., Sasaki, T., 1998. Estimation of time-varying origin-destination flows from traffic counts: A neural network approach. *Mathematical and computer modelling*, 27(9-11), pp.323-334.
- Yang, X., Chang, G.L., 2017. Estimation of Time-Varying Origin–Destination Patterns for Design of Multipath Progression on a Signalized Arterial. *Transportation Research Record*, 2667(1), pp.28-38.
- Yang, X., Cheng, Y., Chang, G.L., 2015. A multi-path progression model for synchronization of arterial traffic signals. *Transportation Research Part C: Emerging Technologies*, 53, pp.93-111.
- Yang, X., Lu, Y., Hao, W., 2017. Origin-destination estimation using probe vehicle trajectory and link counts. *Journal of Advanced Transportation* 2017.
- Yin, X., Wu, G., Wei, J., Shen, Y., Qi, H. and Yin, B., 2021. Multi-stage attention spatial-temporal graph networks for traffic prediction. *Neurocomputing*, 428, pp.42-53.
- Zhang, K., He, Z., Zheng, L., Zhao, L., Wu, L., 2021. A generative adversarial network for travel times imputation using trajectory data. *Computer - Aided Civil and Infrastructure Engineering*, 36(2), pp.197-212.
- Zhang, X., Huang, C., Xu, Y. and Xia, L., 2020, October. Spatial-temporal convolutional graph attention networks for citywide traffic flow forecasting. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (pp. 1853-1862).
- Zhao, Y., Huang, P., Xiao, L., Ma, J., Hu, J., Wang, Y., 2017. A scalable short-term origin-destination estimation approach based on modularized network with connected vehicle technology. In *96th Annual Meeting of the Transportation Research Board (TRB), Washington, DC, USA*.
- Zheng, C., Fan, X., Wang, C., Qi, J., 2020, April. Gman: A graph multi-attention network for traffic prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 34, No. 01, pp. 1234-1241).
- Zhou, X., Mahmassani, H.S., 2006. Dynamic origin-destination demand estimation

- using automatic vehicle identification data. *IEEE Transactions on intelligent transportation systems* 7, 105-114.
- Zhou, X., Mahmassani, H.S., 2007. A structural state space model for real-time traffic origin-destination demand estimation and prediction in a day-to-day learning framework. *Transportation Research Part B:Methodological* 41, 823-840.33
- Zhussip, M., Soltanayev, S. and Chun, S.Y., 2019b. Extending Stein's unbiased risk estimator to train deep denoisers with correlated pairs of noisy images. *Advances in neural information processing systems*, 32, pp.1465-1475.
- Zhussip, M., Soltanayev, S. and Chun, S.Y., 2019a. Training deep learning based image denoisers from undersampled measurements without ground truth and without image prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10255-10264).