STGAN: Spatio-Temporal Generative Adversarial Network for Traffic Data Imputation

Ye Yuan, Yong Zhang, *Member, IEEE*, Boyue Wang, Yuan Peng, Yongli Hu, *Member, IEEE*, and Baocai Yin

Abstract—The traffic data corrupted by noise and missing entries often lead to the poor performance of Intelligent Transportation Systems (ITS), such as the bad congestion prediction and route guidance. How to efficiently impute the traffic data is an urgent problem. As a classic deep learning method, Generative Adversarial Network (GAN) achieves remarkable success in image recovery fields, which opens up a new way for the traffic data imputation. In this paper, we propose a novel spatio-temporal GAN model for the traffic data imputation (STGAN). Firstly, we design the generative loss and center loss, which not only minimizes the reconstructed errors of the imputed entries, but also ensures each imputed entry and its neighbors conform to the local spatio-temporal distribution. Then, the discriminator uses the convolution neural network classifier to judge whether the imputed matrix conforms to the global spatio-temporal distribution. As for the network architecture of the generator, we introduce the skip-connection to keep all well preserved data unchanged, and employ the dilated convolution to capture the spatio-temporal correlation in the traffic data. The experimental results show that our proposed method obviously outperforms other competitive traffic data imputation methods.

Index Terms—Data mining, generative adversarial network, traffic data imputation

1 Introduction

With the rapid growth in vehicles and traffic sensors, Intelligent Transportation Systems (ITS) play a critical role in urban management and control, including congestion relief, route guidance, and event planning. The quality of observed traffic data can significantly determine the effectiveness of traffic data analysis algorithms and then affect the subsequent decision and reliability of ITS. However, noise and missing entries in the observed traffic data have been produced due to hardware malfunction or data transmission. These poor quality data undoubtedly decrease the performance of ITS. Therefore, how to accurately impute the missing entries is an urgent problem.

Traffic data collected from multiple neighbor roads in a certain time can construct a matrix, which naturally contains a strong spatial-temporal correlation. With this conclusion, we can infer the missing entries according to the remaining entries in the observed traffic data matrix. Compared with the commonly-used interpolation method [1], learning based data imputation methods apply the mentioned spatial-temporal correlation and obviously improve the imputation performance, such as Low-Rank Imputation [2], [3], Sparse Regularized SVD (SRSVD) [4], Dynamic Mode Decomposition [5], Kalman Filter [6], Multiple Imputation using Chained Equations (MICE) [7], and MissForest methods [8]. But, their imputation performance drops dramatically as the data missing rate increases.

Fortunately, we notice that deep learning methods are very suitable to impute the data with spatial-temporal correlation [9], [10], [11]. In fact, some researchers have explored this correlation existed in the traffic data [12], [13] and achieved remarkable success, e.g., Artificial Neural Networks (ANN) [14], [15] and Generative Adversarial Network (GAN) [11]. Duan *et al.* [16] treat the traffic data imputation as a one-dimensional vector filling task and use ANN to impute the missing traffic entries of one road.

Compared with ANN, GAN [11] can learn the distribution from the input data and simulate the new data that conforms to above learned distribution, which is more suitable for traffic data imputation tasks. Furthermore, the adversarial mechanism between the generator and discriminator learns the optimal distribution. The generator tries to simulate the values as real as possible to deceive the discriminator, while the discriminator aims to distinguish the fake values from true values. Chen *et al.* [17] initially apply GAN to generate the traffic data, but ignores the local spatio-temporal correlation of the missing entries. To tackle this problem, SpaceGAN [18] is proposed as a generative data augmentation model for geospatial domains, which designs a sampling process taking the spatial structure to preserve the statistical properties, such as the local spatial autocorrelation. Due to the development of

- *Ye Yuan, Yong Zhang, Boyue Wang, Yongli Hu, and Baocai Yin are with the Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing Artificial Intelligence Institute, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China. E-mail: yey@ieee.org, {zhangyong2010, wby, huyongli}@bjut.edu.cn, ybc@dlut.edu.cn.*
- *Yuan Peng is with China Electronics Technology Group, Taiji Company Ltd., Beijing 100124, China. E-mail: yuan.peng@outlook.com.*

Manuscript received 17 May 2021; revised 8 January 2022; accepted 13 February 2022. Date of publication 24 February 2022; date of current version 16 January 2023.
This work was supported in part by the National Key R&D Program of China under Grant 2021ZD0111900, in part by the National Natural Science Foundation of China under Grants 62072015, 61906011, U1811463, U19B2039, and U21B2038, and in part by Beijing Municipal Science and Technology under Grant KM202010005014.
(Corresponding author: Boyue Wang.)
Recommended for acceptance by M. Qiu.
Digital Object Identifier no. 10.1109/TBDATA.2022.3154097

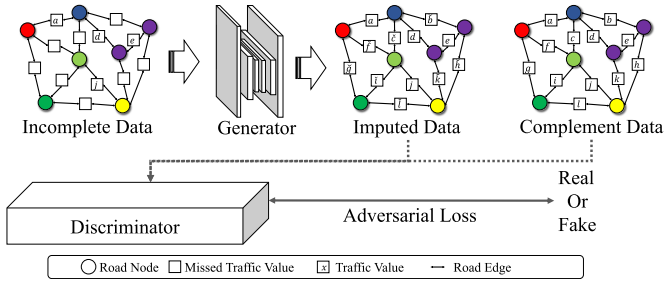


Fig. 1. A brief illustration of STGAN. Given the incomplete traffic data, a residual auto-encoder generates the corresponding imputed traffic data, then a discriminator judges whether the imputed data conforms to the distribution of the real traffic data.

computer vision and GAN, researchers are beginning to consider traffic data as images. Li *et al.* [19] impute the missing values at different missing rates by transforming the traffic data into image data. Yang *et al.* [20] use VGG network to capture the semantic features of the image of the traffic data. But images ignore the features of the traffic itself.

In this paper, we consider the spatio-temporal correlation of missing traffic data and propose an improved GAN-based traffic data imputation method, called spatio-temporal Generative Adversarial Network (STGAN). Fig. 1 briefly shows the whole structure of STGAN. In the generator network, we build an index to record the missing entries in the incomplete matrix, guiding the generator network by focusing on the missing entries. Furthermore, the new network architecture and the generative loss are designed to keep the structure of input data and learn its spatio-temporal information. In the discriminator network, we use a CNN classifier to judge whether the imputed matrix conforms to the global spatio-temporal distribution.

Our main contributions are listed as follows,

- We propose an improved generative adversarial network framework for the traffic data imputation tasks. We introduce GAN to capture a large distribution of traffic data and link the neighboring data captured by the dilated convolutional to optimize the imputation process.
- Inspired by the strong correlation between the traffic data and its surrounding neighbors, we propose the center loss to make the imputed entries conform to the corresponding neighbor distribution. In addition, we design the generative loss to measure the distance between each missing entry and its real value.
- We design two network architectures. Firstly, the skip-connection mechanism is introduced to ensure the observed values unchanged and avoid missing the deep layer features. Secondly, the dilate convolution mechanism is employed to enhance the perception of neighboring regions.
- Through the experimental results, STGAN can obviously improve the effectiveness of traffic imputation in ITS.

The rest of the paper is organized as follows. In Section 2, we review the related work about traffic data imputation and generative methods. In Section 3, we propose a novel GAN method to impute the missing traffic data. In Section 4, we evaluate the proposed method with the Beijing road network dataset. Finally, the conclusion is discussed in Section 5.

2 Related Work

In this section, we firstly review the traffic imputation methods and then introduce the standard generative adversarial network.

2.1 Traffic Data Imputation

Linear interpolation is a basic method for the matrix imputation, which fills in each missing entry by the mean of its neighbors or other linear calculation methods. One common-used method is Historical Interpolation [1], which exploits multiple data points collected from the same period of different days to impute the missing entries. Another commonly used method is K-Nearest Neighbor [21], which calculates the weighted average value of K surrounding neighbors to impute the missing entry. However, K-Nearest Neighbor often performs poorly when some surrounding data points are also lost. The performance of these methods is highly dependent on the quality of the surrounding data of missing entries.

Non-negative Matrix Factorization is introduced to impute the missing entries [22], [23]. Sparse Regularization SVD (SRSVD) [4] decomposes the matrix similar to SVD, which applies the spatio-temporal relationship of traffic data to estimate the missing entries. Another matrix decomposition method, Low Rank Matrix Fitting (LMaFit) [24], employs the low-rank decomposition to imputes the missing entries and constructs the nonlinear continuous excessive relaxation. These matrix factorization methods also adhere to the quality of non-negative matrix, so it lacks scalability and cannot deal with a large number of missing entries.

In recent two years, traffic data imputation researches achieved much progress. Li *et al.* [25] propose a Bayesian Vector Auto-Regression model to handle the unevenly-spaced traffic collision data with missing values. Zhang *et al.* [26] utilize the inherent correlation hidden in traffic data to impute the missing data. Chen *et al.* [27] propose an Augmented Tensor Factorization model. Chen *et al.* [28] extend the Bayesian Probabilistic Matrix Factorization model onto the higher-order tensors to achieve the traffic data imputation tasks. Li *et al.* [29] introduce the multi-view learning to estimate the missing traffic data. Wang *et al.* [3] reconstruct the missing traffic data based on low-rank matrix factorization. Boquet *et al.* [30], [31] design a variational autoencoder to learn the distribution from the missing data and then generate the imputed data from such distribution. Pamula *et al.* [32] estimate the sensitivity of the prediction to the missing traffic data. Zhuang *et al.* [33] transform the raw data into spatial-temporal images and then implement a convolutional neural network on the images to achieve the traffic data imputation tasks. Zhao *et al.* [34] adopt the popular SAE, LSTM and GRU for the imputation task.

2.2 Generative Adversarial Network (GAN)

Generative methods can accurately learn the distribution of the given dataset and then generate several samples that conform to this distribution. Early statistical methods analyze the distribution properties of data well [35], [36], [37], but they are difficult to handle the nonlinear relationship in

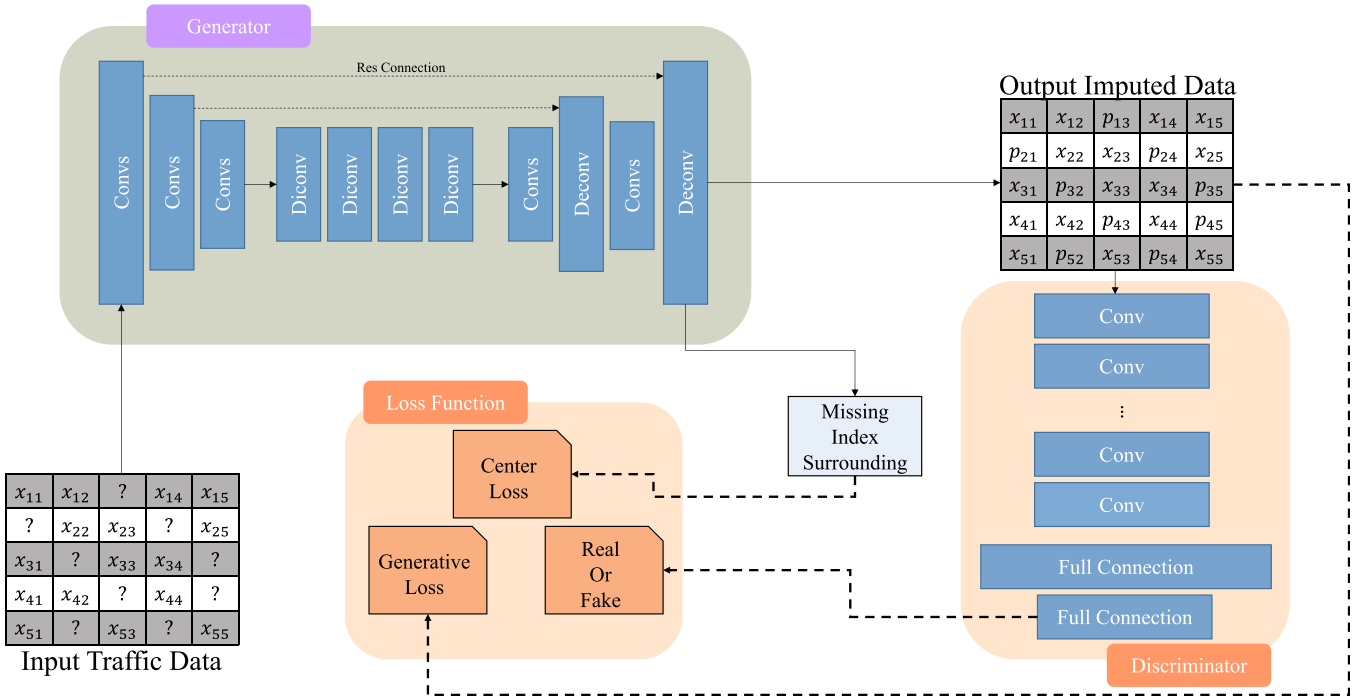


Fig. 2. The STGAN architecture. Given an incomplete traffic matrix, the generator extracts the spatio-temporal features using dilated convolutions, and outputs the imputed complete matrix. The discriminator judges whether the imputed complete matrix conforms to the distribution of the real data. The center loss and generative loss can supervise the neighbor correlation of missing entries. The dotted lines in the generator mean the skip-connection, and the other dotted lines mean the direction of the loss function.

complex data. Fortunately, Generative Adversarial Network (GAN) [11] is proposed and successfully applied in the data imputation tasks. Yoon *et al.* [38] design the full-connection generator and discriminator to recover the MNIST images.

GAN contains two main modules: the generator G and discriminator D. Specifically, the generator G aims to make the generated data $G(\mathbf{Z})$ conform to the distribution of real data p_{data}, while the discriminator D judges whether the generated data $G(\mathbf{Z})$ is real or false. Overall, the training procedure is a two-player min-max game with the following objective function:

$$\begin{aligned} J^D &= -E_{\mathbf{C}\sim p_{data}} \log D(\mathbf{C}) - E_{\mathbf{Z}\sim p_{noise}} \log\left(1 - D(G(\mathbf{Z}))\right), \\ J^G &= -E_{\mathbf{Z}\sim p_{noise}} \log D(G(\mathbf{Z})), \end{aligned} \tag{1}$$

where p_{data} denotes the distribution of real data and p_{noise} means the distribution of noise. $\mathbf{C}$ denote the real data and $\mathbf{Z}$ represents the random noise, respectively.

GAN has been creatively introduced into the field of data imputation due to its wonderful performance on the data simulation. GLCIC [39] uses the global and local context discriminators to capture the globally and locally consistent information of one image, respectively, ensuring the consistency and details in the imputed image data. GAIN [38] prompts the auto-encoder generators to impute the missing image data and provides a Hint Matrix to the discriminator. MisGAN [40] builds two generators to simulate the distribution of missing data and impute the complex high-dimensional missing data, respectively. CollaGAN [41] converts the image imputation problem to a multi-domain image-to-image translation task so that the single generator and discriminator can successfully estimate the missing image data using the remaining clean dataset.

3 GAN for Traffic Data Imputation

In this section, we firstly introduce the data imputation problem, and then describe the improved GAN-based traffic data imputation method.

3.1 Problem Formation

The incomplete matrix can be modeled as the combination of a complete matrix and a missing index as follow,

$$\mathbf{Z} = (\mathbf{1}\mathbf{1}^T - \mathbf{M}) \odot \mathbf{C}, \tag{2}$$

where $\mathbf{1}\mathbf{1}^T$ is an all-one matrix. $\mathbf{C}$ denotes a complete matrix, and $\mathbf{Z}$ represents its corresponding incomplete matrix. $\mathbf{P}$ is defined as the corresponding imputed matrix of $\mathbf{Z}$. Index $\mathbf{M}$ marks the position of missing data. Operator $\odot$ means the element-wise multiplication. So, if $\mathbf{M}_{i,j} = 1$ then the entry $C_{i,j}$ is missing. According to formula (2), our goal is to impute the given incomplete matrix $\mathbf{Z}$ and get the corresponding imputed matrix $\mathbf{P}$.

As we know, the incomplete traffic data and the corresponding complete traffic data are difficult to be simultaneously captured in the natural environment. In order to simulate the incomplete traffic data, we randomly select a fraction of the entries and remove them from the complete data as formula (2). Details in Section 4.

3.2 The Proposed Method

As a standard deep learning method, GAN is commonly used in data imputation tasks due to the adversarial learning strategy. GAN constrains the imputed data to conform the distribution of the real data. Considering the spatio-temporal information in the traffic data, we propose a novel GAN-based traffic data imputation method (STGAN).

Fig. 2 exhibits the main architecture of STGAN. Given an incomplete matrix, the generator outputs a complete matrix without missing values. Then, the discriminator verifies whether the imputed matrix is real or false. Here, we choose the improved auto-encoder and CNN classifier as the generator and discriminator, respectively. In addition, a series of loss functions are designed from multiple aspects to enhance the performance of imputing the incomplete data, including the Adversarial loss, Generative loss, and Center loss.

Adversarial Loss. In the adversarial loss, the generator maximally fools the discriminator while the discriminator maximally distinguishes the real and imputed data, which encourages the imputed data conform to the global data distribution. The adversarial loss is defined as,

$$\begin{aligned} J^D &= -E_{\mathbf{C}\sim\mathcal{C}}\log D(\mathbf{C}) - E_{\mathbf{Z}\sim\mathcal{I}}\log\left(1 - D(G(\mathbf{Z}))\right), \\ J^G &= -E_{\mathbf{Z}\sim\mathcal{I}}\log D(G(\mathbf{Z})), \end{aligned} \tag{3}$$

where G and D represent the generator and discriminator, respectively. J^D and J^G denote the corresponding objective functions. E is the expectation function. The incomplete traffic data $\mathbf{Z}$ is the input data of the generator, and the complete matrix $\mathbf{C}$ denotes the real data for the discriminator. Finally, $\mathcal{C}$ and $\mathcal{I}$ contain a set of complete and incomplete traffic data matrices.

Generative Loss. For the imputed matrix $\mathbf{P} = G(\mathbf{Z})$, we adopt the least square regularizer to measure the reconstructed error between the imputed value and its real value of each missing entry, which ensures the reconstructed capability of the generator G,

$$J_{gen} = \sum_{r\in\mathcal{R}} (P^r - C^r)^2, \tag{4}$$

where the missing index $\mathcal{R}$ records the positions of all missing traffic entries. So, P^r and C^r mean the r-th entry in the imputed matrix $\mathbf{P}$ and the real data $\mathbf{C}$, respectively. J^r measures the reconstruction error of the r-th entry. Unlike the standard GAN method, above loss function considers the reconstructed error of the missing entries in the training stage, which helps the generator to impute the missing entries more accurately in practical applications.

Center Loss. To further exactly evaluate the missing traffic entries, each imputed entry and its corresponding neighbor entries should conform to the local distribution. We exploit KL-divergence to measure the neighbor correlation of each missing entry. Therefore, the center loss function can be defined as,

$$J_{center} = -\sum_{r\in\mathcal{R}} \sum_{i,j=1}^{N\times N} \hat{C}^r_{i,j} \log \frac{\hat{P}^r_{i,j}}{\hat{C}^r_{i,j}}, \tag{5}$$

where $\hat{\mathbf{P}}^r \in R^{N\times N}$ and $\hat{\mathbf{C}}^r \in R^{N\times N}$ denote the surrounding matrices of entry P^r and entry C^r, respectively. $\hat{P}^r_{i,j}$ and $\hat{C}^r_{i,j}$ denote the (i,j)-th entry in matrices $\hat{\mathbf{P}}^r$ and $\hat{\mathbf{C}}^r$, respectively. These values are normalized between 0 and 1. N is the radius of the neighbor region of the missing entry, which is set as $N = 2$ in our experiments. It should be noted that each missing entry is recorded by the missing index $\mathcal{R}$. To acquire $\hat{P}^r$ and $\hat{C}^r$, we use the missing mask to locate all missing points. For each missing point, a small matrix is obtained with the radius N, so-called the neighbor matrix, which help the proposed model to capture the local information around the missing points.

Final Objective Loss Function. Now, we combine above loss functions to obtain the final objective loss function of STGAN as follows,

$$\mathcal{L}_{Object} = J^D + J^G + J_{gen} + J_{center} \tag{6}$$

- J^D and J^G derive from the standard GAN model. Minimizing these two metrics makes the imputed data sufficiently conform to the distribution of the real data. In other words, the imputed data looks realistically.
- J_{gen} measures the loss between the imputed data and the corresponding real data. Minimizing this metric makes the imputed data sufficiently accurate.
- J_{center} constrains each imputed value conform to the distribution of its surrounding neighboring values. Minimizing this metric makes the imputed data conform to its local distribution.

First, we back-propagate the values of loss functions (J^G, J_{gen}, and J_{center}) to optimize the generator. The optimization algorithm makes the traffic matrix imputed by the generator realistic, accurate and consistent with the spatio-temporal relationship. Then, we back-propagate the value of loss function J^D to optimize the discriminator, so that the optimized discriminator can distinguish the matrix imputed by the generator is true or not. We alternately optimize the generator and the discriminator. By optimizing this adversarial process, STGAN can impute the realistic and accurate traffic matrices.

After training the STGAN model, we can exploit the trained network to impute the missing traffic entries. This network can achieve traffic data imputation tasks almost in real time.

4 Experimental Results

In this section, we design a series of experiments to test the performance of STGAN.

4.1 Datasets and Settings

Dataset. We evaluate STGAN on the traffic data collected from the Third Ring Expressway in Beijing called the Beijing Road dataset and the subway network in Beijing called the Beijing Subway dataset. The traffic data in the Road dataset records consist of the average speed data, which is captured from the floating cars with GPS onboard. These records gather 9760 roads from July to October in 2016, totally 61296 time intervals. The Beijing Subway dataset collects from the swipe data of personal bus cards, which covers all Beijing metro lines from July to August in 2015. The swipe data is captured every five minutes on more than 300 stations in total. All datasets are fully observed and we randomly chose twenty percent of the dataset as the testing set and the rest as the training set. We randomly chose twenty percent of the dataset as the testing set and the rest as the training set.

Since the traffic data exists the strong spatial and temporal correlation (where and when the data is collected), we organize the traffic data into the form of multi-dimensional

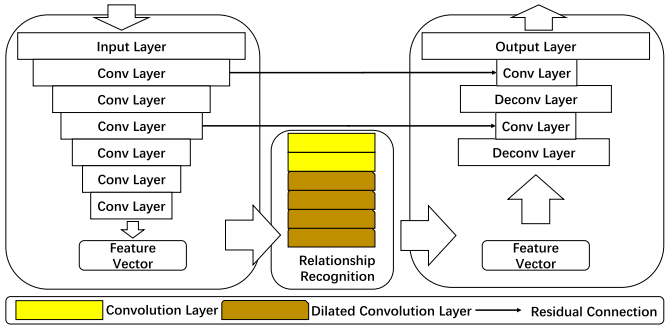


Fig. 3. The generator architecture. The incomplete traffic data is encoded into the high-dimension feature vectors. Then, the dilated convolution extracts the spatio-temporal information from those feature vectors. Finally, the feature vectors are decoded into the imputed traffic matrix. In addition, we utilize skip connection to keep the well preserved data unchanged. The arrow lines mean in the figure the skip-connection and the big arrows mean data transfer.

matrix [17], [29]. In each matrix, we organize the whole time units of one road into one column, and multiple roads forms the matrix according to their spatial relationships (road segment $\times$ time).

Missing Mask. It is impossible to simultaneously capture the complete and incomplete data of the same road in the practical environment. Therefore, we have to regard the observed traffic data as the complete data. Then we randomly discard some entries as formula (2) to construct the incomplete data. The goal of STGAN is to use the remaining data to infer the missing entries.

The mask matrix M can be generated according to the following three cases,

- Missing completely at random (MCAR): We randomly and evenly throws a set of entries from the complete matrix.
- Missing over time: In a certain period, no data is detected or the detector is in trouble, then several rows in the matrix are missing.
- Missing over space: GPS detectors in taxi cars cannot cover some sections, then several columns in the matrix are missing.

Above constructed dataset includes pairs of matrices. Each pair consists of one incomplete matrix and its corresponding complete matrix.

Network Architecture. In order to better achieve traffic data imputation tasks, we design a novel encoder-decoder architecture as the generator shown in Fig. 3. The main components of the network architecture are: 1) Six convolution layers firstly extract the feature vector of the incomplete traffic data. 2) To focus on the spatio-temporal correlation hidden in traffic data, four dilated convolution layers [42] and the two convolution layers are constructed. 3) Finally, the feature vector is translated into the imputed traffic data through the convolution and de-convolution layers. It is worth noting that the skip-connection after each de-convolution are used to retain the integrity information in the imputed traffic data.

In the generator network, we select RELU as the activate function in each convolutional layer. Each convolutional layer chooses a (3,3) kernel function and sets both stride and padding as 1. But, four dilated convolutional layers respectively set the padding as $2, 4, 8$ and 16, and other settings are same with previous convolutional layers. In addition, each deconvolutional layer selects the kernel function size of (4,4), stride 1, padding 0, and connects to an average pooling layer. We use an index matrix to record all missing position before training the model, which avoids the misoperation to the missing data. For all missing entries, we fill in 0.

In the discriminator network, the output of the generator is taken as the input of this discriminator. Similarly, we select ReLU as the activate function in each convolutional layer. But each convolutional layer uses the kernel size of (5,5), stride 1 and padding 2. Finally, a sigmoid function is chosen to output the result after the fully connected layer.

In addition, batch size, learning rate and the number of epochs are set as 8, 0.001 and 100, respectively.

Settings. Testing missing rate denotes the percentage of missing entries in the testing set. For example, the testing missing rate 0.3 denotes that thirty percent of entries in the incomplete matrix are missing. In order to cover various testing missing rate events in practical data imputation applications, The testing missing rate varies from 0.2 to 0.8 to test the robustness of the data imputation methods.

To assess all methods form different aspects, we compare the imputed matrix P against the complete matrix C, and calculate their standard metrics: Root Mean Squared Error (RMSE) and positive MAE and negative MAE [43] as follows:

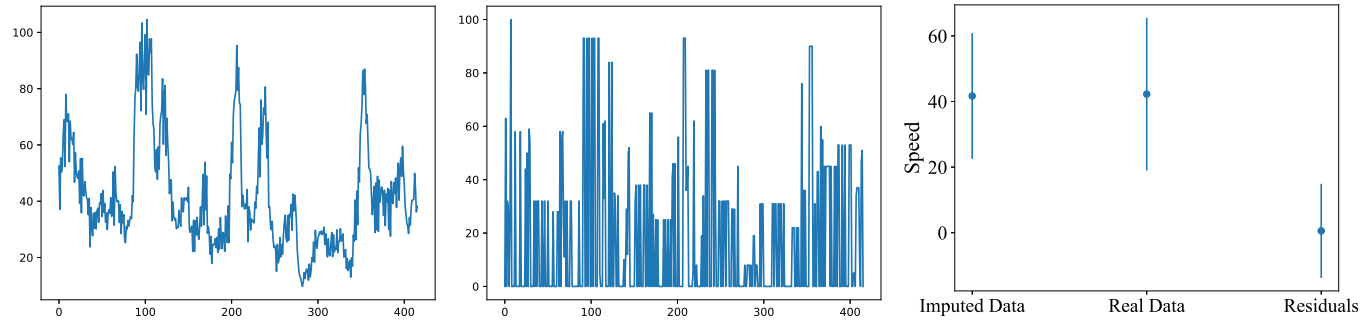
$$
\begin{aligned}
NMAE &= \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} \frac{|P_{i,j} - C_{i,j}|}{\overline{C}} \\
RMSE &= \sqrt{\frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} (P_{i,j} - C_{i,j})^2} \\
MAE+ &= \frac{\sum_{i=1}^{W} \sum_{j=1}^{H} max(imputed_{i,j} - GroundTruth_{i,j}, 0)}{W \times H}
\end{aligned}
\tag{7}
$$

where W and H are the number of rows and columns of the matrix. $\overline{C}$ denotes the mean value of matrix C. Obviously, NMAE is the percentage.

4.2 STGAN Model Validity

We attempt to verify the model validity through a serious of experiments in this subsection. We randomly select a road interval to show its traffic data in Fig. 4. Specifically, Fig. 4a represents the curve of the imputed data; Fig. 4b shows the curve of the corresponding real data; Fig. 4c shows that the mean-std deviation of the imputed data, the real data and the residuals between the real and imputed data. As shown in Fig. 4, the mean deviation of the imputed and real data are very close to each other, and the mean residuals between them are close to 0; therefore, the imputed data should be accurate.

We train multiple STGAN models on various training sets with the training missing rates varied from 0.2 to 0.8. Then, we test these models on the testing sets with different testing missing rates. All experimental results are shown in Tables 1 and 2. The trained STGAN performs well when the testing missing rate closes to the training missing rate, while its performance declines as the testing missing rate leaves away the training missing rate. Giving test data with lower



(a) Imputed Data (b) Real Data (Ground Truth) (c) The mean-std deviation of (a)-(c)

Fig. 4. We exhibit the time series of traffic data of one road on the Beijing Road dataset. The imputed data is shown in (a). The real data is shown in (b). The x-axis represents time and the y-axis represents traffic speed. Then, (c) shows that the mean-std deviation of the imputed data, the real data and the residuals between the real and imputed data. (from left to right).

TABLE 1
Parameter Analysis (NMAE) on the Beijing Road Dataset. Training Missing Rate and Testing Missing Rate are Abbreviated to Training and Missing. We Validate the Performance of Various Combinations of Training Missing Rate and Test Missing Rate

Training/Testing	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.2	0.105	0.107	0.117	0.142	0.187	0.275	0.416
0.3	0.161	0.169	0.182	0.211	0.256	0.336	0.451
0.4	0.160	0.161	0.163	0.172	0.194	0.253	0.361
0.5	0.163	0.168	0.169	0.174	0.184	0.218	0.300
0.6	0.217	0.205	0.199	0.198	0.198	0.201	0.206
0.7	0.309	0.282	0.255	0.233	0.213	0.210	0.253
0.8	0.337	0.306	0.274	0.251	0.231	0.220	0.223

TABLE 2
Parameter Analysis (RMSE) on the Beijing Road Dataset. Training Missing Rate and Testing Missing Rate are Abbreviated to Training and Missing

Training/Testing	0.2	0.3	0.4	0.5	0.6	0.7	0.8
0.2	5.998	6.259	6.811	8.051	10.234	14.374	20.269
0.3	8.561	8.823	9.286	10.519	12.451	15.798	20.392
0.4	8.468	8.583	8.624	9.076	10.097	12.793	17.418
0.5	8.696	8.960	8.911	9.142	9.534	11.312	15.128
0.6	10.657	10.175	9.929	9.928	9.949	10.122	10.511
0.7	14.651	13.603	12.416	11.475	10.665	10.606	12.813
0.8	15.768	14.657	13.340	12.353	11.500	11.060	11.274

missing rate to the model with higher training missing rate, the performance of the model is still lower than under the training missing rate due to the different features of the input data

Observing 2, our model performs relatively well when its training missing rate equals to the intrinsic missing rate of the testing data. The practical missing rate needs the long-term statistical information to determine. However, the road conditions change rapidly, which leads to the quality of collected data also changes rapidly. So choosing a relative robust model that performs well under various missing rates can reduce the computation complexity. In Fig. 5, our model receives the relative good results around the testing missing rate 0.6 due to the training missing rate of the selected model is 0.6. Base on above evaluation results, the overall performance is satisfied when the training missing rate is 0.6.

We also evaluate the performance of STGAN under the case of the mixed missing rate (the percentage of each missing rate is equal). We compare it with the STGAN model with the training missing rate = 0.6 in Figs. 10a, 10b and 10c. It can be seen that the STGAN with the training missing rate 0.6 obviously outperforms the STGAN with the mixed training missing rate, which owes to the uneven distribution of features.

4.3 Comparison to Existing Methods

To demonstrate the improvements brought by STGAN, we compare STGAN with several classic traffic data imputation methods, including Generative Adversarial Nets (GAIN) [38], Multiple Imputation using Chained Equations (MICE) [7], Traffic flow imputation using parallel data and generative adversarial networks (TFIPDGAN) [12], Artificial Neural Networks (ANN), ImageCNN [33], SpaceGAN [18]. SpaceGAN [18] is a generative data augmentation model for geospatial domains, which designs a sampling process taking the spatial structure to preserve the statistical properties, such as the local spatial autocorrelation.

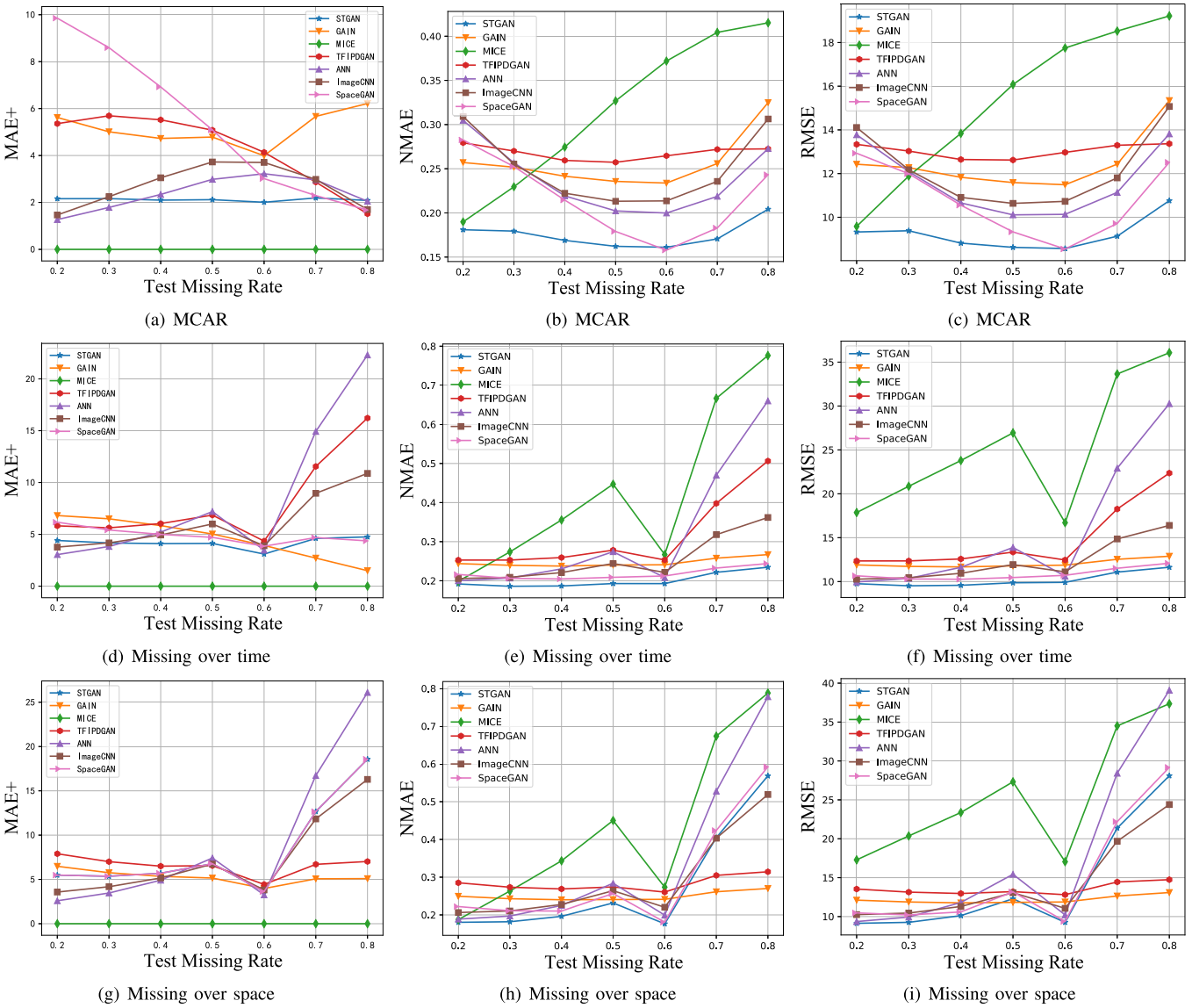


Fig. 5. The comparative experimental results in three types of data missing cases on the Beijing Road dataset.

The matrix size used in our model (e.g., 416×416) is significantly larger than other machine learning based traffic imputation methods (e.g., 30×30). The full-connection layer structure is difficult to handle the large size data due to the amount number of parameters and the high computation source requirements, so we have to replace the full-connection layers in the compared methods (GAIN, ANN, TFIPDGAN) by the convolutional layers. The layer stack architecture is still retained.

Figs. 5 and 6 exhibit the whole experimental results. In MCAR case, STGAN far exceeds other methods, which means the spatio-temporal features are remained and the imputed entries obey the distribution of the well preserved data. In time case, STGAN still outperforms other methods, but the advantage of STGAN becomes decreases. The temporal features are difficult to be extracted due to we miss the excessive time series, which verifies the importance of temporal features. In space case, STGAN performs better than other methods at relative low missing rates (<0.6), but performs worse than some methods at high missing rates (>0.6). The structure information of the whole road network is lost so much that it is counterproductive to capture the spatio features. SpaceGAN performs relatively well among the comparison methods, especially when the training and testing missing rates are the same. However, the performance decreases substantially as the gap between the training and test missing rates increases. Our method is generally superior to SpaceGAN, especially in terms of the robustness and stability. We believe the proposed method captures the surrounding neighborhood relations and considers the global information of the whole traffic area. STGAN is most sensitive to data features in MCAR case and remains stable in time and low missing-rate space cases, but in the high missing-rate space case there are negative results due to the large loss of road network structure. The center loss of STGAN focuses on the extraction of neighboring road features, and the spatial missing affects the performance of the model more seriously compared to the temporal missing. Figs. 10a, 10b and 10c can further support this viewpoint.

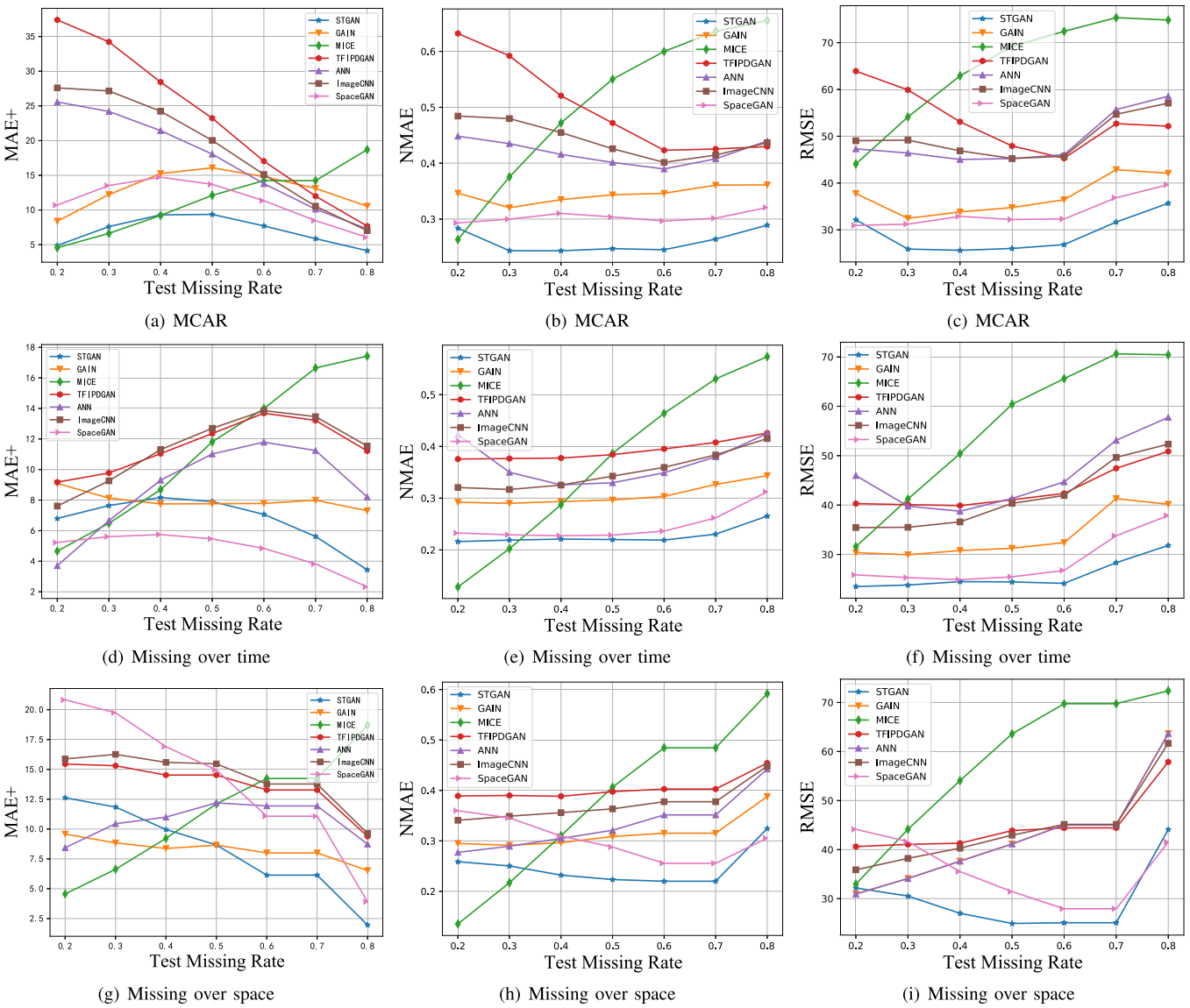


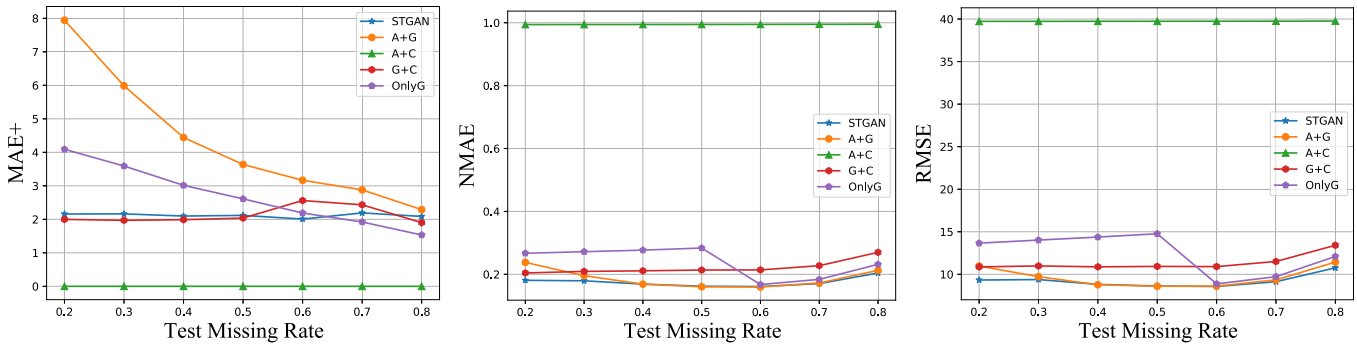
Fig. 6. The comparative experimental results in three types of data missing cases on the Beijing Subway dataset.

4.4 Ablation Analysis

Our objective function consist of three modules: adversarial loss (A), generative loss (G) and center loss (C). In order to fairly evaluate the effectiveness and advantage of each module in STGAN, we design a set of ablation methods as follows,

- A+G: This method contains the normal adversarial loss and generative loss, which is designed to prove that the neighbor correlation constrained by the center loss module is beneficial to the traffic data imputation.
- G+C: This method consists of the generative loss and center loss, constructing the basic generator of STGAN. In other words, this method lacks of discriminator, which is designed to demonstrate the advantage of the adversarial learning.
- A+C: This method is consists of the adversarial loss and the center loss, which is designed to verify the performance of single GAN.
- Only G: This method only retain the generative loss.

Figs. 7a, 7b and 7c display the experimental results of all ablation methods. STGAN obtains the optimal values of both RMSE and NMAE, which is superior to other ablation methods. The RMSE value of STGAN at testing missing rate 0.2 is 0.18, which is remarkably lower than A+G. We believe that the neighbor correlation can be mostly preserved at the low testing missing rates, so the information residual of the neighbor roads is relatively complete. STGAN applies above neighbor information to make the imputed entries more accurate. Therefore, the center loss function is necessary in our proposed methods. The performance of G+C is much worse than A+G and STGAN under various testing missing rate conditions, which attributes to the effect of adversarial loss function. In other words, the discriminator can handle the global distribution of data. In addition, the experimental result of A+C proves that the single GAN works bad for the traffic data imputation tasks due to it cannot keep the neighbor correlation of the input data in the imputed data. Only G performs poorly as the testing



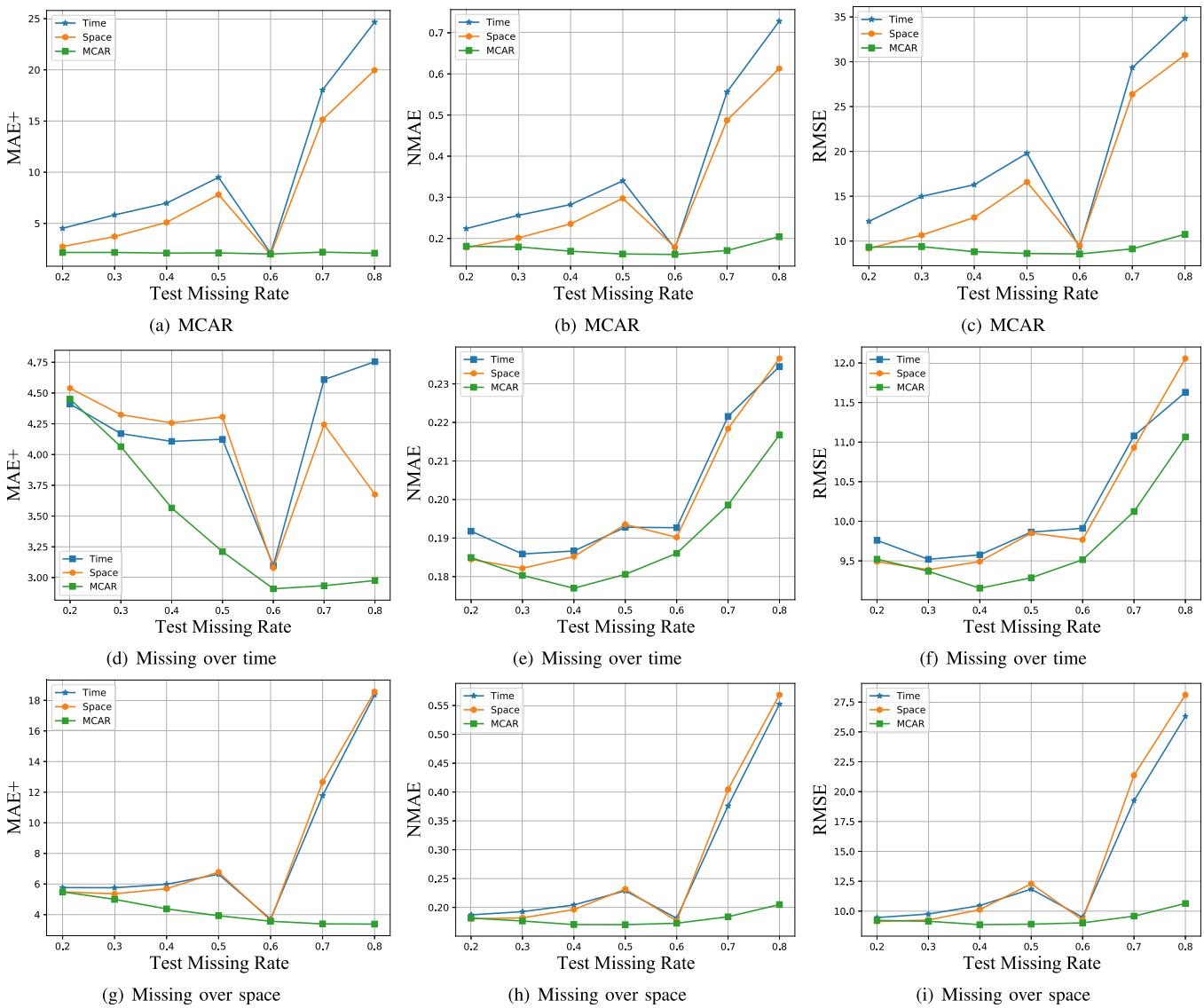
(a) Ablation (MAE+) (b) Ablation (NMAE) (c) Ablation (RMSE)

Fig. 7. Sub-figures (a)-(c) show the ablation experiment results of STGAN on the Beijing Road dataset.

missing rate varying from 0.2 to 0.5, which captures less neighbor correlation of the missing data. But it works better than G+C when the testing missing rate varies from 0.6 to 0.8, indicating that the large number of missing entries makes the neighbor correlation difficult to extract.

4.5 Robustness Analysis

In order to verify the performance of STGAN with the different data missing cases, we conduct a set of robustness validity experiments as follow, includes three training settings and the corresponding test cases:

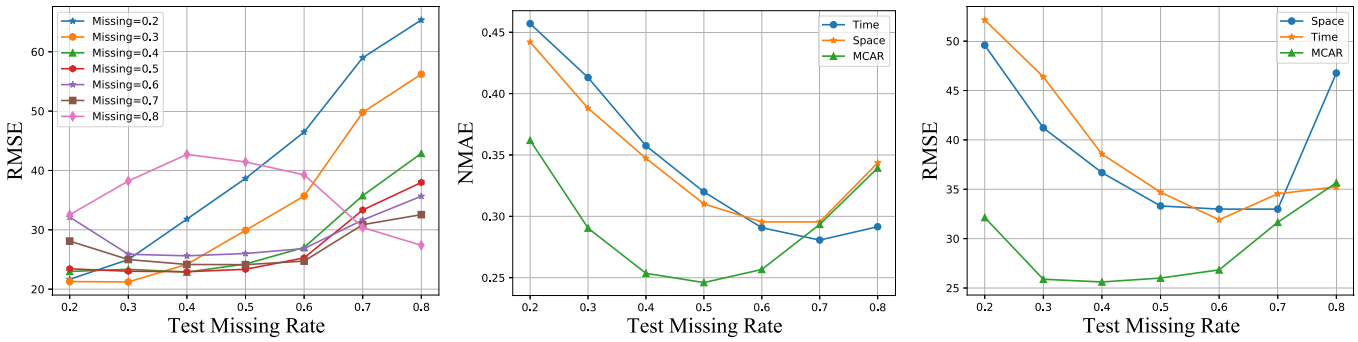


(a) MCAR (b) MCAR (c) MCAR

(d) Missing over time (e) Missing over time (f) Missing over time

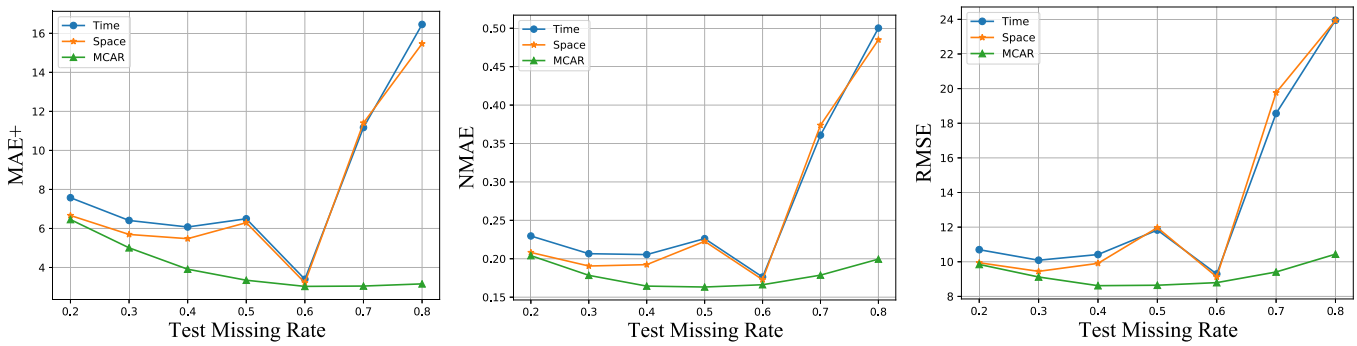
(g) Missing over space (h) Missing over space (i) Missing over space

Fig. 8. The performance of the STGAN model trained on each data missing case on the Beijing Road dataset.



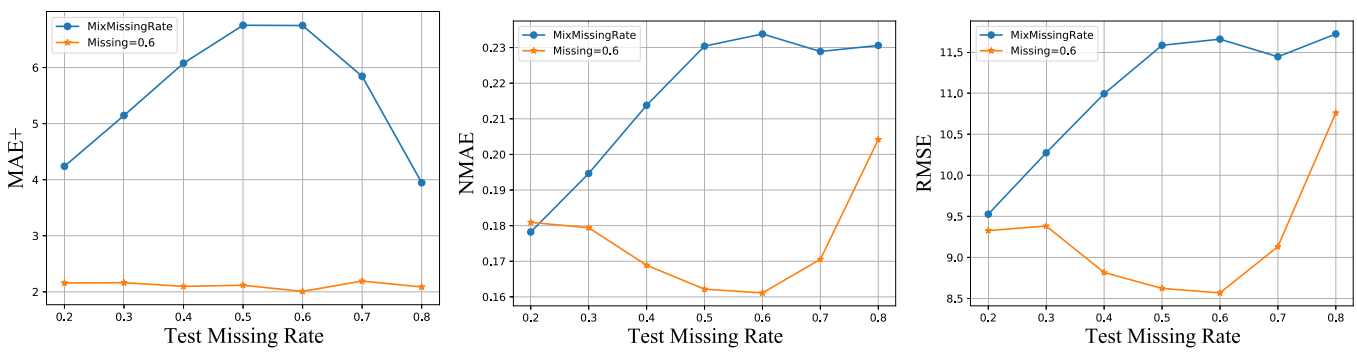
(a) Missing rate setting (b) Missing case setting (c) Missing case setting

Fig. 9. The experimental results on the Beijing Subway dataset. (a) shows the performance of STGAN under different missing rate settings. (b) and (c) shows the performance of the model trained under MCAR for other missing cases(Space, Time) data.



(a) MAE+ (b) NMAE (c) RMSE

Fig. 10. The STGAN model is trained on a mix of three cases of missing data on the Beijing Road dataset, then tested on each case of missing data, respectively.



(a) MAE+ Mixed Missing Rate (b) NMAE Mixed Missing Rate (c) RMSE-Mixed Missing Rate

Fig. 11. The testing results of different missing rates on the Beijing Road dataset. The orange line denotes the STGAN model is trained on the data with the missing rate 0.6, and the blue line is the STGAN model is trained on the data with the mixed missing rate.

- The model is trained according to the MCAR case, and tests data missing by the time and space cases.
- The model is trained according to the time missing case, and tests data missing by the MCAR and space cases.
- The model is trained according to the space missing case, and tests data missing by the time and MCAR cases.

As shown in the Figs. 8 and 9, the STGAN model trained by the MCAR case performs stable in other two cases. The STGAN model trained by the time missing case and the STGAN model trained by the space missing case perform well in the MCAR case. The STGAN trained by the time missing case has a slight degradation in the space missing case, which may be the difference of the modes in different cases. The performance of the STGAN trained by the space missing case in the time missing case is still stable when the testing missing rate is low, but it does not perform well when the loss rate is high. It is difficult to extract local and global spatio-temporal at a high space testing missing rate.

In MCAR case, STGAN far exceeds other methods, which means that the spatio-temporal features are remained and the imputed entries obey the distribution of the well preserved data. In time case, STGAN still outperforms other methods, but the advantage of STGAN becomes decreases. The temporal features are difficult to be extracted due to we

miss the excessive time series, which verifies the importance of temporal features. In space case, STGAN performs better than other methods at relative low missing rates (<0.6), but performs worse than some methods at high missing rates (>0.6). The structure information of the whole road network is lost so much that it is counterproductive to capture the spatial features. STGAN is sensitive to the data features in MCAR case and remains the stable in time and low missing-rate space cases, but in the high missing-rate space case there are negative results due to the large loss of road network structure. Figure. 8 and 9 can further support this viewpoint. Therefore, in view of the stability of the STGAN, we choose the MCAR model as the general form of STGAN.

To further test the robustness of STGAN, we compare the model trained by the mixture of three missing cases (percentage of each case data is equally) with the models trained by three missing cases alone. As shown in Fig. 10a and 10c, STGAN gets the best performance at the MCAR missing case, while the time missing case and the space missing case get the similar performance. STGAN gets the best performance on the MCAR missing case, while it obtains the similar performance on the time missing case and the space missing case. It can be considered that the data distribution of the mixed case is closer to the MCAR case.

5 Conclusion

In this paper, a novel GAN-based method is proposed to impute the incomplete traffic data, which is named as STGAN. The generator learns the spatio-temporal correlation hidden in traffic data and then imputes the missing entries, and the discriminator judges whether the imputed data is real or false. The adversarial learning strategy is beneficial to enhance the data imputation quality. In addition, the designed generative loss and center loss consider the neighbor correlation of missing entries. The trained network can impute the new incomplete traffic data in real time, which can be employed in practical applications. In further researches, we will introduce the attention mechanisms to further capture the spatial-temporal correlations of traffic data.

References

[1] J. Chen and J. Shao, "Nearest neighbor imputation for survey data," *J. Official Statist.*, vol. 16, no. 2, pp. 113–132, 2000.

[2] H. Tan, G. Feng, J. Feng, W. Wang, Y.-J. Zhang, and F. Li, "A tensor-based method for missing traffic data completion," *Transp. Res. Part C: Emerg. Technol.*, vol. 28, pp. 15–27, 2013.

[3] Y. Wang, Y. Zhang, X. Piao, H. Liu, and K. Zhang, "Traffic data reconstruction via adaptive spatial-temporal correlations," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1531–1543, Apr. 2019.

[4] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," *Proc. ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 4, pp. 267–278, 2009.

[5] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *J. Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[6] A. C. Harvey, *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge, U.K.: Cambridge Univ. Press, 1990.

[7] I. R. White, P. Royston, and A. M. Wood, "Multiple imputation using chained equations: Issues and guidance for practice," *Statist. Med.*, vol. 30, no. 4, pp. 377–399, 2011.

[8] D. J. Stekhoven and P. Bühlmann, "Missforest non-parametric missing value imputation for mixed-type data," *Bioinformatics*, vol. 28, no. 1, pp. 112–118, 2012.

[9] L. Qu, L. Li, Y. Zhang, and J. Hu, "PPCA-based missing data imputation for traffic flow volume: A systematical approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 512–522, Sep. 2009.

[10] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proc. ICML Workshop Unsupervised Transfer Learn.*, 2012, pp. 37–49.

[11] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[12] Y. Lv, Y. Chen, L. Li, and F.-Y. Wang, "Generative adversarial networks for parallel transportation systems," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 3, pp. 4–10, 2018.

[13] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with Big Data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[14] M. Zhong, S. Sharma, and P. Lingras, "Genetically designed models for accurate imputation of missing traffic counts," *Transp. Res. Rec.*, vol. 1879, no. 1, pp. 71–79, 2004.

[15] Y. Duan, Y. Lv, Y.-L. Liu, and F.-Y. Wang, "An efficient realization of deep learning for traffic data imputation," *Transp. Res. Part C: Emerg. Technol.*, vol. 72, pp. 168–181, 2016.

[16] Y. Duan, Y. Lv, W. Kang, and Y. Zhao, "A deep learning based approach for traffic data imputation," in *Proc. IEEE Conf. Intell. Transp. Syst.*, 2014, pp. 912–917.

[17] Y. Chen, Y. Lv, and F.-Y. Wang, "Traffic flow imputation using parallel data and generative adversarial networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1624–1630, Apr. 2020.

[18] K. Klemmer, A. Koshiyama, and S. Flennerhag, "Augmenting correlation structures in spatial data using deep generative models," 2019, *arXiv:1905.09796*.

[19] H. Li, Y. Wang, and M. Li, "Modified GAN model for traffic missing data imputation," in *Proc. CICTP*, 2020, pp. 3013–3023.

[20] B. Yang, Y. Kang, Y. Yuan, X. Huang, and H. Li, "ST-LBAGAN: Spatio-temporal learnable bidirectional attention generative adversarial networks for missing traffic data imputation," *Knowl.-Based Syst.*, vol. 215, 2021, Art. no. 106705.

[21] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.

[22] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 556–562.

[23] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[24] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *Math. Program. Comput.*, vol. 4, no. 4, pp. 333–361, 2012.

[25] Z. Li, H. Yu, G. Zhang, and J. Wang, "A Bayesian vector autoregression-based data analytics approach to enable irregularly-spaced mixed-frequency traffic collision data imputation with missing values," *Transp. Res. Part C: Emerg. Technol.*, vol. 108, pp. 302–319, 2019.

[26] H. Zhang *et al.*, "Missing data detection and imputation for urban anpr system using an iterative tensor decomposition approach," *Transp. Res. Part C: Emerg. Technol.*, vol. 107, pp. 337–355, 2019.

[27] X. Chen, Z. He, Y. Chen, Y. Lu, and J. Wang, "Missing traffic data imputation and pattern discovery with a Bayesian augmented tensor factorization model," *Transp. Res. Part C: Emerg. Technol.*, vol. 104, pp. 66–77, 2019.

[28] X. Chen, Z. He, and L. Sun, "A Bayesian tensor decomposition approach for spatiotemporal traffic data imputation," *Transp. Res. Part C: Emerg. Technol.*, vol. 98, pp. 73–84, 2019.

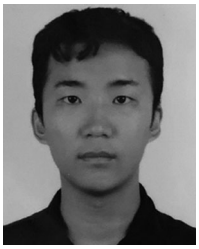
[29] L. Li, J. Zhang, Y. Wang, and B. Ran, "Missing value imputation for traffic-related time series data based on a multi-view learning method," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 8, pp. 2933–2943, Aug. 2019.

[30] G. Boquet, A. Morell, J. Serrano, and J. L. Vicario, "A variational autoencoder solution for road traffic forecasting systems: Missing data imputation, dimension reduction, model selection and anomaly detection," *Transp. Res. Part C: Emerg. Technol.*, vol. 115, 2020, Art. no. 102622.

[31] G. Boquet, J. L. Vicario, A. Morell, and J. Serrano, "Missing data in traffic estimation: A variational autoencoder imputation method," in *Proc. ICASSP IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 2882–2886.

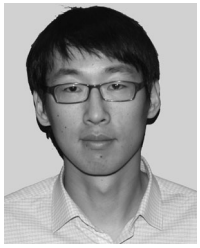
[32] T. Pamuła, "Impact of data loss for prediction of traffic flow on an urban road using neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 1000–1009, Mar. 2019.

[33] Y. Zhuang, R. Ke, and Y. Wang, "Innovative method for traffic data imputation based on convolutional neural network," *IET Intell. Transport Syst.*, vol. 13, no. 4, pp. 605–613, 2018.
[34] J. Zhao, Y. Nie, S. Ni, and X. Sun, "Traffic data imputation and prediction: An efficient realization of deep learning," *IEEE Access*, vol. 8, pp. 46713–46722, 2020.
[35] V. Mnih *et al.*, "Generating more realistic images using gated MRF's," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 2002–2010.
[36] N. Le Roux, N. Heess, J. Shotton, and J. Winn, "Learning a generative model of images by factoring appearance and shape," *Neural Comput.*, vol. 23, no. 3, pp. 593–650, 2011.
[37] H. Shim, "Probabilistic approach to realistic face synthesis with a single uncalibrated image," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3784–3793, Aug. 2012.
[38] J. Yoon, J. Jordon, and M. Van DerSchaar, "Gain: Missing data imputation using generative adversarial nets," 2018, *arXiv:1806.02920*.
[39] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–14, 2017.
[40] S. Cheng-Xian Li, B. Jiang, and B. Marlin, "MisGAN: Learning from incomplete data with generative adversarial networks," 2019, *arXiv:1902.09599*.
[41] D. Lee, J. Kim, W.-J. Moon, and J. C. Ye, "CollaGAN: Collaborative gan for missing image data imputation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2487–2496.
[42] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.
[43] Y. Yu, Y. Zhang, S. Qian, S. Wang, Y. Hu, and B. Yin, "A low rank dynamic mode decomposition model for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 10, pp. 6547–6560, Oct. 2021.

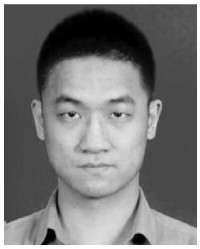


Ye Yuan received the BS degree in software engineering from the Taiyuan University of Technology, Taiyuan, China, in 2016. He is currently working toward the postgraduation degree with the Beijing University of Technology. His current research interests include machine learning, probabilistic model, traffic data restoration, and computer vision.

Yong Zhang received the ME degree from Shandong University, Jinan, China, in 2004, and the PhD degree from the Beijing University of Technology, Beijing, China, in 2010. He is currently an associate professor with the Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing Advanced Innovation Center for Future Internet Technology, Faculty of Information Technology, Beijing University of Technology. His research interests include computer graphics, virtual reality, and traffic simulation. He is a member of CCF.



Boyue Wang received the BSc degree in computer science from the Hebei University of Technology, China, in 2012, and the PhD from the Beijing University of Technology, Beijing, China, in 2018. He is currently a postdoctor with the Beijing Municipal Key Laboratory of Multimedia and Intelligent Software Technology, Beijing University of Technology. His current research interests include computer vision, pattern recognition, manifold learning, and kernel methods.



Yuan Peng received the MS degree in software engineering and IT methods applied to business management from the University of Picardy Jules Verne, France, in 2011 and 2012. He is currently a senior engineer with China Electronics Technology Group. His current research interests include geographic information system, air traffic control, computer graphics, atmospheric operation mode, and radar echo.

Yongli Hu received the PhD degree from the Beijing University of Technology in 2005. He is currently a professor with the Advanced Innovation Center of Future Internet Technology, Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Faculty of Information Technology, Beijing University of Technology, Beijing, China. His research interests include computer graphics, pattern recognition, and multimedia technology.

Baocai Yin received the MS and PhD degrees in computational mathematics from the Dalian University of Technology, Dalian, China, in 1988 and 1993, respectively. He is currently a professor with the Department of Electronic Information and Electrical Engineering, Dalian University of Technology. He is also the director of the Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing, China. He has authored or coauthored more than 200 academic papers in prestigious international journals, including *IEEE Transactions on Multimedia* and *IEEE Transactions on Circuits and Systems for Video Technology*, and top-level conferences, such as Infocom and ACM SIGGRAPH. His research interests include multimedia, image processing, computer vision, and pattern recognition. Dr. Yin is currently an editorial member of the *Journal of Information and Computational Science*.