Transportation Research Part C 34 (2013) 108–120

Contents lists available at SciVerse ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Efficient missing data imputing for traffic flow by considering temporal and spatial dependence

Li Li, Yuebiao Li, Zhiheng Li *

Department of Automation, Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, Beijing 100084, China

ARTICLE INFO

Article history:
Received 3 December 2012
Received in revised form 11 March 2013
Accepted 22 May 2013

Keywords:
Traffic flow
Missing data
Temporal and spatial dependence
Probabilistic principle component analysis (PPCA)
Kernel probabilistic principle component analysis (KPPCA)

ABSTRACT

The missing data problem remains as a difficulty in a diverse variety of transportation applications, e.g. traffic flow prediction and traffic pattern recognition. To solve this problem, numerous algorithms had been proposed in the last decade to impute the missed data. However, few existing studies had fully used the traffic flow information of neighboring detecting points to improve imputing performance. In this paper, probabilistic principle component analysis (PPCA) based imputing method, which had been proven to be one of the most effective imputing methods without using temporal or spatial dependence, is extended to utilize the information of multiple points. We systematically examine the potential benefits of multi-point data fusion and study the possible influence of measurement time lags. Tests indicate that the hidden temporal–spatial dependence is nonlinear and could be better retrieved by kernel probabilistic principle component analysis (KPPCA) based method rather than PPCA method. Comparison proves that imputing errors can be notably reduced, if temporal–spatial dependence has been appropriately considered.

1. Introduction

The quality of traffic service depends on the accuracy and completeness of the information that we collected (Sharma et al., 2004). However, the missing data problem remains a challenge in many current traffic operation systems (Turner et al., 2000; Chen et al., 2002; Bickel et al., 2007).

Despite the fast growing reliability of traffic data collection and transmission systems, the missing of traffic flow data is still notable in many places. For example, at hundreds of detection points within PeMS traffic flow database (PeMS), more than 5% of data are lost. Such missing data problem significantly impedes further applications, because most existing analyzing methods rely on complete data (at least after imputation) (Vlahogianni et al., 2004; van Lint et al., 2005; Zhang et al., 2011; Chen et al., 2012).

An extensive variety of missing data imputing methods had been proposed in the last decade to solve this problem (Smith et al., 2003; Zhong et al., 2004a). From the viewpoint of modeling philosophy, we can roughly divide these methods into three kinds: prediction based, interpolation based and statistical learning based methods (Qu et al., 2009; Tan et al., 2013).

Prediction based methods directly adopted existing traffic prediction methods, including Autoregressive Integrated Moving Average (ARIMA) (Ahmed and Cook, 1979) and Feed-Forward Neural Network (FFNN) (Vlahogianni et al., 2005; Karlaftis and Vlahogianni, 2011). In these approaches, a missing datum is viewed as a value to predict and is then forecasted based on the relationship identified from historical past-to-future data pairs (Zhong et al., 2004b). However, two major differences between traffic flow prediction and imputing had not been well considered in such approaches. First, many prediction

* Corresponding author. Address: #808, Central Main Building, Tsinghua University, Beijing 100084, China. Tel.: +86 (10) 62795503.
E-mail address: zhhli@mail.tsinghua.edu.cn (Z. Li).

0968-090X/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.trc.2013.05.008

models do not fully utilize the data collected after the missed data, which would possibly degrade imputing performance. Second, if a consecutive series of data are all lost, the prediction methods often fail to give satisfactory results.

Interpolation-based methods fill the missing data with a weighted average of known data that is either temporal-neighboring (collected from the same detector at the same time period but in neighboring days) or pattern-neighboring (collected from the same detector at the same time period but in other days with similar daily flow variation patterns) (Zhong et al., 2005; Yin et al., 2012). Such approaches highly depend on the assumption that daily traffic flows have strong similarity in a few days. However, this assumption sometimes fails in practice. Nevertheless, interpolation methods do not explicitly describe the stochastic variations of traffic flow data and may miss some useful information other than daily flow profile.

Differently, statistical learning based methods try to take advantages of the statistic feature of traffic flow. The key postulate of such approaches is assuming a special probability distribution that is followed by the observed data. Then, the values that best fits the assumed probability distribution will be taken as the missed data (Ni and Leonard, 2005; Ni et al., 2005; Qu et al., 2009; Li et al., in press).

Though statistical learning based methods often suppose strong hypothesis over traffic flow data, their imputing performance is often better than conventional models. This is mainly because the assumed probability distribution captures the essentials of traffic flow variations. Considering both imputing accuracy and running speed, probabilistic principle component analysis (PPCA) based imputing method was found to be one most effective imputing methods for single detector cases (Qu et al., 2009; Li et al., in press). Moreover, the values imputed by PPCA method are statistically consistent with the distribution of original traffic flow data. Via robust principle component analysis, outliers or unusual patterns in the data can also be easily detected, when a component causes a notable increase in covariance.

Although several missing data imputing methods had been proposed, there are still several important challenges that remain to be thoroughly answered. As pointed out in Karlaftis (2012), one of them is how to appropriately consider temporal and spatial dependence to boost imputing performance.

As well known, the variations of traffic flow at neighboring points are tightly related, since parts of traffic move along the roads and pass these points sequentially. Given the knowledge of flow at upstream points would more or less benefit flow predict/reconstruction at a downstream point, if proper methods are applied.

There has been a growing concern on traffic flow prediction using the information collected from multiple points in transportation networks. For example, the multi-variable state-space model had been tested in Stathopoulos and Karlaftis (2003). Different extended ARIMA models were studied in Der Voort et al. (1996), Williams (2001), Kamarianakis and Prastacos (2003), Min et al. (2010) and Min and Wynter (2011). Various Bayesian networks were designed and examined in Sun et al. (2006), Ramezani et al. (2010) and Sun and Xu (2011). Recently, the adaptive least absolute shrinkage and selection operator (LASSO) was introduced to perform model selection and coefficient estimation simultaneously (Kamarianakis et al., 2012; Sun et al., 2012).

However, online system identification for these prediction models is challenging, when we encountered consecutive missing data. One possible solution is to apply non-parametric models, e.g. the kernel regression model recently proposed in Haworth and Cheng (2012). But few existing multi-point prediction methods allow us conveniently utilize the information collected both before and after the missed data. Besides, when massive data from thousands of points are considered, the training time costs in some algorithms will be a great burden, too.

To upgrade imputing performance, we compare PPCA method and kernel probabilistic principle component analysis (KPPCA) model based method on multiple point information fusion in this paper. Consistent with existing reports on traffic flow prediction regarding spatial dependence, our tests indicate that the hidden temporal–spatial dependence appears nonlinear. Therefore, KPPCA model behaves more powerfully, since it relaxes the linear mapping assumption of PPCA model. In agreement with Haworth and Cheng (2012), comparison proves that kernel based learning models better describe the nonlinear spatial–temporal dependence around neighboring points. This allows a well-tuned KPPCA model outperforms a PPCA model.

To give a detailed analysis, the rest of this paper is arranged as follows. Section 2 first briefly introduces PPCA and KPPCA based missing data imputing method and explains the ideas behind them. Section 3 discusses how to incorporate spatial dependence into PPCA and KPPCA methods. Section 4 presents the imputing testing results and account for why KPPCA method better recover the hidden dependence between traffic flow data recorded at neighboring detecting points. Finally, Section 5 concludes the whole paper.

2. PPCA based missing data imputing methods without using spatial dependence

2.1. PPCA based missing data imputing

PPCA based missing data imputing for traffic flow had been well discussed in Qu et al. (2009) and Li et al. (in press). In this paper, we just give a brief review of its formulation.

PPCA model (Tipping and Bishop, 1999; Roweis and Ghahramani, 1999; Ilin and Raiko, 2010) supposes each sample y_i depends on a q-dimensional latent variable x_i as

$$y_i = Wx_i + \mu + \varepsilon_i \tag{1}$$

where we usually require $d \ll q$ to retrieve the hidden common feature of traffic flow data. μ is a d-dimensional column vector that characterizes the sample average of y_i. Here, the subscript i denotes the index of the observation/latent variable.

PPCA model assumes that the latent variables x_i follows a q-dimensional multivariate Gaussian distribution, $x_i \sim N_q(0, I)$, because the fluctuations of traffic flow time series roughly follow a Gaussian distribution (Stathopoulos and Tsekeris, 2006; Chen et al., 2012). The d-dimensional column vector ε_i is introduced as isotropic noise satisfying $\varepsilon_i \sim N_d(0, \sigma^2 I)$, where σ^2 is scaling factor. It relaxes the strict assumption on daily-flow similarity and makes the model more flexible.

The projection matrix $W \in R^{d\times q}$ represents a mapping between the latent variable space and the observed variable space that is followed by all the observed-latent variable pairs (y_i, x_i). Indeed, model (1) assume that different elements of a particular y_i are tightly dependent, since they are mainly determined by different projections of the same hidden variable x_i. That is, for $y_i(j_1)$ and $y_i(j_2)$, the j_1th and j_2th elements of vector y_i, we have

$$y_i(j_1) = W_{j_1} x_i + \mu(j_1) + \varepsilon_i(j_1), \quad y_i(j_2) = W_{j_2} x_i + \mu(j_2) + \varepsilon_i(j_2) \tag{2}$$

where $\mu(j_1)$, $\mu(j_2)$, $\varepsilon_i(j_1)$, $\varepsilon_i(j_2)$ are the corresponding elements of vector μ and ε_i. W_{j_1} and W_{j_2} are the j_1th and j_2th rows of projection matrix W.

When all data y_i are available (not missing), given the value of q, we take the maximum likelihood estimators of μ, W and σ^2 according to the above probability distribution assumptions. More precisely, we maximize the following log-likelihood function in terms of μ, W and σ^2

$$\arg\max_{\mu,W,\sigma^2} L_{c0} = \arg\max_{\mu,W,\sigma^2} \ln\left\{\prod_{i=1}^{n} p(y_i|\mu, W, \sigma^2)\right\} \tag{3}$$

where

$$p(y_i|\mu, W, \sigma^2) = \int p(y_i|x_i, \mu, W, \sigma^2) p(x_i) dx_i \tag{4}$$

$$p(y_i|x_i, \mu, W, \sigma^2) = N(y_i|Wx_i + \mu, \sigma^2 I), p(x_i) = N(x_i|0, I) \tag{5}$$

Due to the Gaussian distribution assumption, Problem (3) is a convex optimization problem and can be analytically solved. It was shown in Tipping and Bishop (1999) that the solution is achieved when W spans the principal sub-space of the data, which therefore gives the name of probabilistic principal component analysis.

When some elements of $y_1, \ldots, y_n$ are missing (say y_{i_miss} and y_{i_obse} are the missing part and observed part of y_i, respectively), we still search μ, W and σ^2 that produce maximum likelihood in agreement with the known data

$$\arg\max_{\mu,W,\sigma^2} L_{c1} = \arg\max_{\mu,W,\sigma^2} \sum_{i=1}^{n} \ln\{p(y_{i\_miss}, y_{i\_obse}, x_i)\} \tag{6}$$

where the conditional probability density function is

$$\begin{aligned} p(y_{i\_miss}, y_{i\_obse}, x_i) &= p(y_i, x_i|\mu, W, \sigma^2) = p(y_i|x_i, \mu, W, \sigma^2) \times p(x_i) \\ &= (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{\|y_i - Wx_i - \mu\|^2}{2\sigma^2}\right\} \times (2\pi)^{-q/2} \exp\left\{-\frac{1}{2} x_i^T x_i\right\} \end{aligned} \tag{7}$$

Meanwhile, we impute y_{i_miss} to best fit with the above distribution assumptions and thus the estimated maximum likelihood. In practice, we directly determine μ as

$$\mu = \left[\frac{1}{n}\sum_{i=1}^{n} y_i(1), \quad \ldots, \quad \frac{1}{n}\sum_{i=1}^{n} y_i(d)\right]^T \tag{8}$$

where $y_i(y)$ is the yth element of ith sample. If some data are missing, μ is calculated by taking the average of the available data.

Expanding (7), we can see that the objective function is a nonconvex function in terms of W, σ^2, μ and y_{i_miss}. According to Boyd and Vandenberghe (2004), problem (6) is a not a convex optimization problem and cannot be analytically solved. In this paper, we resort to the Expectation Maximization (EM) algorithm that is widely used in statistics and learning theory (Dempster et al., 1977; McLachlan and Krishnan, 2008). In short, EM algorithm solve (6) via two steps iteratively.

In the kth iteration of E-Step of EM algorithm, we update our guess of the missing part of y_i, $i = 1, \ldots, n$, based on current estimation of W and σ^2 as

$$\tilde{y}_i(k) = W(k)[M(k)^{-1} W(k)^T (y_i(k-1) - \mu)] + \mu \tag{9}$$

where $M(k) \in R^{q\times q}$ in the covariance matrix calculated as:

$$M(k) = W(k)^T W(k) + \sigma^2(k) I \tag{10}$$

In the kth iteration of M-Step, with the newly guessed $\tilde{y}_i(k)$, we conversely update our guess of W and σ^2 via maximum likelihood estimator as

$$\begin{aligned} W(k+1) &= S(k)W(k)(\sigma(k)^2 I + M(k)^{-1}W(k)^T S(k)W(k))^{-1} \\ \sigma(k+1)^2 &= \frac{1}{d}tr(S(k) - S(k)W(k)M(k)^{-1}W(k+1)^T) \end{aligned} \tag{11}$$

where

$$S(k) = \frac{1}{n}\sum_{i=1}^{N}(\tilde{y}_i(k) - \mu)(\tilde{y}_i(k) - \mu)^T \tag{12}$$

We repeat these operations until we will reach a local maximum of the likelihood function given in (6). That is, the estimations of W and σ^2 converge to fixed values. The final guess of the missing part of y_i will then be imputed. Please check Tipping and Bishop (1999), Roweis and Ghahramani (1999) and Ilin and Raiko (2010) for detailed EM algorithm to PPCA model.

2.2. KPPCA based missing data imputing

Although PPCA method gives impressive missing data imputing results (Qu et al., 2009; Li et al., in press), researchers are now fathoming whether the above simple linear mapping model in (2) prevents us from even better performance. To answer this question, KPPCA based imputing method was studied in Li et al. (in press).

In short, KPPCA assumes a nonlinear relationship between an observed sample y_i and a latent variable x_i (Schölkopf et al., 1997; Lawrence, 2005; Sanguinetti and Lawrence, 2006)

$$\phi(y_i) = Wx_i + \varepsilon_i \tag{13}$$

where $\phi(\cdot):R^d \to R^k$ is an image that maps the observation into an special nonlinear unknown space (feature space) with a higher dimension, i.e. $d < k$. $x_i \sim N_q(0, I)$ and $\varepsilon_i \sim N_k(0, \sigma^2 I)$ have similar meanings as PPCA model but different dimensions.

In KPPCA model, we further assume the prior distribution of $W \in R^{k\times q}$ as

$$p(W) = \prod_{i=1}^{d} N_k(w_i|0, I) \tag{14}$$

where w_i is the ith row of W.

Using kernel tricks, we do not need to determine the exact formulation of $\phi(\cdot)$. Instead, we define a certain kernel function $k(\alpha, \beta):R^d \times R^d \to R$ can be expressed as a special inner product. For example, the Gaussian kernel $k(\alpha, \beta) = \exp(-\gamma\|\alpha - \beta\|^2)$, $\gamma > 0$, is a typical radial basis kernel function. Thus, given any two observation samples y_i, y_j, we can directly calculate their inner product value

$$k(y_i, y_j) := \langle \varphi(y_i), \varphi(y_j) \rangle, i, j = 1, \ldots, n \tag{15}$$

Integrating all the observation samples, we can define a $n \times n$ kernel matrix K, whose elements are $K_{ij} = k(y_i, y_j)$. Then, we extract the principal components within feature space.

Suppose, we order the eigenvalues (from big to small) and the corresponding eigenvectors of the kernel matrix K as λ_i and $\alpha_i \in R^n$, $i = 1, \ldots, n$. We define the feature variable F on the first q largest eigenvalues and variance of noise σ^2 on the rest eigenvalues as

$$F = diag(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_q}) \times (\alpha_1, \alpha_2, \ldots, \alpha_q)^T \tag{16}$$

$$\sigma^2 = \frac{1}{n-q}\sum_{i=q+1}^{n}\lambda_i \tag{17}$$

In KPPCA, we still aim to impute y_{i_miss} to best fit with the estimated maximum likelihood and the above distribution assumption (Sanguinetti and Lawrence, 2006; Lawrence, 2005). After certain derivation, we finally optimize the following likelihood function in terms of y_{i_miss}

$$\arg\max_{y_{i\_miss}} L_{c2} = \arg\max_{y_{i\_miss}} \left\{ \frac{1}{2}\ln|B| + \frac{1}{2}\mathrm{tr}(B^{-1}K) \right\} \tag{18}$$

where

$$B = F^T F + \sigma^2 I \tag{19}$$

In this paper, we make an initial guess of the missing data and then iteratively execute the following two steps to impute the missing data:

The First Step: We use the currently estimated missing data to calculate the kernel matrix of K via Eq. (15). Then, we calculate the feature variable F and variance of noise σ^2 from Eqs. (16) and (17). At last, we update our estimation on B via Eq. (19). Then the likelihood function Eq. (18) is updated. If the newly obtained likelihood is nearly the same as the previous one (in other words, our estimation algorithm converges), we stop; otherwise, we go to the second step.
The Second Step: We update our estimation on the missing data in K to optimize Eq. (18) by using the Scaled Conjugate Gradient (SCG) algorithm (Moller, 1993). Then, we go back to the first step. Detailed analysis of SCG can be found in Moller (1993) and is thus omitted here.

2.3. Use PPCA/KPPCA for single point traffic flow data imputing

Suppose we evenly collect traffic flow data at one detector in the first day and get a series of samples as $t^1(1),\ldots,t^1(N)$, where N denotes the number of sample points per day. For example, if the sample time interval is 300 s, we have $N = 288$. In a symbol $t^x(j)$, j is the index of sample point in a day and x is the index of sample day.

Similarly, after recording traffic flow data for D consecutive days, we can get some n-dimensional row vectors, e.g. $T^1 = [t^1(1), t^1(2),\ldots,t^1(N)],\ldots,T^d = [t^D(1), t^D(2),\ldots,t^D(N)]$. If we put these row vectors together, we can get a data matrix

$$T = \begin{bmatrix} T^1 \\ T^2 \\ \vdots \\ T^D \end{bmatrix} = \begin{bmatrix} t^1(1) & t^1(2) & \cdots & t^1(N) \\ t^2(1) & t^2(2) & \cdots & t^2(N) \\ \vdots & \vdots & \ddots & \vdots \\ t^D(1) & t^D(2) & \ldots & t^D(N) \end{bmatrix} \tag{20}$$

PPCA model does not attack $T^1,\ldots,T^D$ directly. Instead, it studies the column vector of the data matrix T and takes each column of T as a new D-dimensional sample vector y_i that appears in model (1). As shown in Fig. 1, we then have N samples denoted as $y_1,\ldots,y_n$, where $y_i = [\,t^1(i) \quad t^2(i) \quad \ldots \quad t^D(i)\,]^T$, $i = 1,\ldots,N$. So in PPCA model, we set $d = D$ and $n = N$, (the sample dimension in PPCA is the row number of T; while the sample number in PPCA is the column number of T).

PPCA mode assumes that the traffic flow values at the same sampling time but in different days are implicitly correlated. From the viewpoint of statistical learning, it indeed presumes that all the elements of a particular y_i follow a special joint

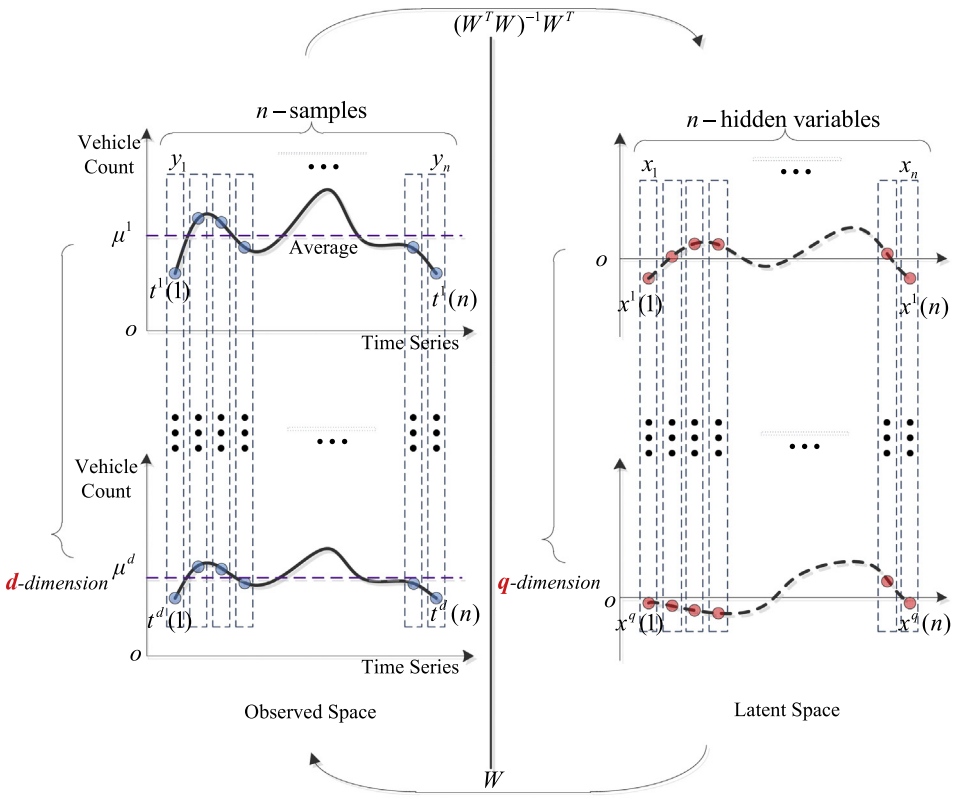

Fig. 1. An illustration of PPCA model.

distribution. This *a priori* knowledge provides insights into traffic flow dynamics other than similarity and thus improves imputing performance.

Previous tests show that linear mapping Gaussian distribution model assumption (1) captures the essentials of traffic flow dynamics (Qu et al., 2009; Li et al., in press). As a result, PPCA based method provides faster reconstruction speed and smaller imputing errors than most conventional methods, when only the information of one point is used.

KPPCA method inherits the merits of PPCA method and is expect to achieve better results. Differently, we take each row of T as a sample vector y_i that appears in KPPCA model (20), leading $y_i = [\,t^i(1) \quad t^i(2) \quad \dots \quad t^i(N)\,]^T$. That is, we set $d = N$ and $n = D$ in KPPCA model (the sample dimension in KPPCA is the column number of T; while the sample number in KPPCA is the row number of T). In other words, KPPCA mode assumes the traffic flow vectors collected in different days are nonlinearly correlated.

Purely from the viewpoint of data modeling, there is no technical difference between treating the column as the sample vector and treating the row as the sample vector, since we can equivalently view them as samples in high dimensional spaces. If we use the column format, we assume a linear mapping relation between the observations (the values of different days) at the same time point as Eq. (1). If we use the row format, we assume a linear mapping relation between the observations (the whole-day data segment) for different days.

Moreover, we can treat the raw format as a special dual PPCA model. As pointed out in Lawrence (2005), both PPCA and dual PPCA belong to special cases of the more general class of Gaussian process-latent variable models and can be handled in a similar way. Any interested readers can find detailed discussions in Lawrence (2005). Constraints by length limits, we will not further explain them in this paper.

As shown in Li et al. (in press), if the historical and real-time information of only one point is used, KPPCA yield slightly better imputing performance than PPCA. However, if we make a trade off among model accuracy, simplicity, and running speed, PPCA is still recommended rather than KPPCA for single point missing data imputing, mainly due to Occam's razor.

We apply the above sample vector y_i defining strategy (in short, column for PPCA and row for KPPCA) to all PPCA and KPPCA models that will be mentioned in the rest of this paper.

For presentation convenience, we abbreviate the above PPCA and KPPCA based models that do not consider neighboring points as **PPCA_1** and **KPPCA_1** in the below.

Besides, the latent variable space dimension d characterizes the implicitly assumed degree of similarity between traffic flow data gotten in different days. We will discuss how to select the best latent variable space dimension in Section 4.2.

As shown in Shawe-Taylor and Cristianini (2004), we usually cannot prove the existence of nonlinear dependence by directly calculating correlation of two variables, especially when the dimensions of data vector are large. So, we take the so called try-and-test procedure to check the potential nonlinear dependence among data. In other words, we will directly examine whether KPPCA works better than PPCA in missing data imputing to verify the necessity of introducing KPPCA model.

In Li et al. (in press), we compared PPCA and KPPCA based missing data imputing methods, using the information of one detector. Results indicated that there is no use to introduce KPPCA for one detector cases. While in the rest of this paper, our focus is the potential benefits of using the information of neighboring detectors.

3. PPCA based missing data imputing methods using temporal–spatial dependence

3.1. Multiple-point PPCA and KPPCA models without time lag

Suppose we evenly collect traffic flow data at m different points in the first day and get a series of sample data vectors as $T_1^1 = [t_1^1(1), t_1^1(2), \dots, t_1^1(N)], \dots, T_m^1 = [t_m^1(1), t_m^1(2), \dots, t_m^1(N)]$. Similarly, after recording for D consecutive days, we can get some N-dimensional row vectors, e.g. $T_1^j = [t_1^j(1), t_1^j(2), \dots, t_1^j(N)], \dots, T_m^j = [t_m^j(1), t_m^j(2), \dots, t_m^j(N)] \quad .j = 1, \dots, N$.

One straightforward modification is to put all these N-dimensional row vectors together and modify the data matrix in (1) as

$$\hat{T} = \begin{bmatrix} T_1^1 \\ T_1^2 \\ \vdots \\ T_1^D \\ \vdots \\ T_m^1 \\ T_m^2 \\ \vdots \\ T_m^D \end{bmatrix} = \begin{bmatrix} t_1^1(1) & t_1^1(2) & \dots & t_1^1(N) \\ t_1^2(1) & t_1^2(2) & \dots & t_1^2(N) \\ \vdots & \vdots & \ddots & \vdots \\ t_1^D(1) & t_1^D(2) & \dots & t_1^D(N) \\ \hline & & \vdots & \\ \hline t_m^1(1) & t_m^1(2) & \dots & t_m^1(N) \\ t_m^2(1) & t_m^2(2) & \dots & t_m^2(N) \\ \vdots & \vdots & \ddots & \vdots \\ t_m^D(1) & t_m^D(2) & \dots & t_m^D(N) \end{bmatrix} \begin{matrix} \\ \\ \\ \text{Detector 1} \\ \\ \\ \\ \\ \text{Detector m} \end{matrix} \tag{21}$$

Similarly, we can directly adopt PPCA and KPPCA model to impute the missing values in $\hat{T}$. For presentation convenience, we abbreviate models working on $\hat{T}$ as **PPCA_2** and **KPPCA_2** in the rest of this paper.

Particularly, **PPCA_2** model takes each column of $\widehat{T}$ as a $(m \times D)$-dimensional sample vector. Thus, we have N samples as $y_1,\ldots,y_N$, $y_i = [\,t_1^1(i) \quad \ldots \quad t_1^D(i) \quad \ldots \quad t_m^1(i) \quad \ldots \quad t_m^D(i)\,]^T$, $i = 1,\ldots,N$. Clearly, the new models operating on $\widehat{T}$ presume implicit correlations between the traffic flow values observed at the same sampling time but different detecting points, since the elements of a particular y_i follow a special joint distribution. Our objective is to characterize this special joint distribution and hence recover the possible relationships between the measured traffic flow values at different points, which finally boosts imputing performance.

Similarly, **KPPCA_2** model takes each row of $\widehat{T}$ as a N-dimensional sample vector and we now get $(m \times D)$ samples.

The larger we set m, the richer information we may employ to estimate the missed data. However, this may also lead to more complex models and longer calculating time. It had been shown that the correlation degrees among different points decrease significantly with respect to distances. So, in conjunction with Kamarianakis et al. (2012), we only consider $m = 3$ in this paper. That is, only the upstream and downstream neighboring detecting points are studied.

For convenience, we denote the upstream detector as *Detector 1*, the middle one (the one used to compare imputing errors) as *Detector 2*, while the downstream one as *Detector 3* in the below. We will only compare **PPCA_2** and **KPPCA_2**'s imputing errors at *Detector 2* with **PPCA_1** and **KPPCA_1**'s imputing errors at *Detector 2*.

3.2. Multiple-point PPCA and KPPCA models with time lag

The above models **PPCA_2** and **KPPCA_2** do not consider the possible delayed correlations between two neighboring points. However, it sometimes argued that the most valuable information from a neighboring point is not its most recent record but its previous/next record; because traffic flow usually spends a certain time to move from one point to another.

One straightforward modification is to add more rows into the sample data matrix. Following the suggestions given in Kamarianakis et al. (2012), we only consider the data sampled one-step ahead at the upstream detector and the data sampled one-step behind at the downstream detector in this paper, since these values are mostly related. Therefore, for each day, we simultaneously add five data vectors into the data matrix.

Particularly, for the jth day, these two vectors for *Detector 1* are

$$\widetilde{T}_1^j = \begin{bmatrix} t_1^j(0) & t_1^j(1) & \ldots & t_1^j(N-1) \\ t_1^j(1) & t_1^1(2) & \ldots & t_1^j(N) \end{bmatrix} \tag{22}$$

where $t_1^j(0)$ is the last sample entry of $(j-1)$th day's traffic flow vector, $t_1^j(0) = t_1^{j-1}(N)$. If $j = 1$, we treat $t_1^j(0)$ as 0.

Likewise, for the jth day, the two vectors for *Detector 3* are

$$\widetilde{T}_3^j = \begin{bmatrix} t_3^j(1) & t_3^j(2) & \ldots & t_3^j(N) \\ t_3^j(2) & t_3^j(3) & \ldots & t_3^j(N+1) \end{bmatrix} \tag{23}$$

where $t_3^j(N+1)$ is the first sample entry of $(j+1)$th day's traffic flow vector, i.e. $t_3^j(N+1) = t_3^{j+1}(1)$. If $j = D$, we treat $t_3^j(N+1)$ as 0.

If we put all these row vectors together, the new data matrix turns out to be

$$\widetilde{T} = \begin{bmatrix} \widetilde{T}_1^1 \\ \vdots \\ \widetilde{T}_1^D \\ T_2^1 \\ T_2^2 \\ \vdots \\ T_2^D \\ \widetilde{T}_3^1 \\ \vdots \\ \widetilde{T}_3^D \end{bmatrix} = \begin{bmatrix} t_1^1(0) & t_1^1(1) & \ldots & t_1^1(N-1) \\ t_1^1(1) & t_1^1(2) & \ldots & t_1^1(N) \\ \vdots & \vdots & \ddots & \vdots \\ t_1^D(0) & t_1^D(1) & \ldots & t_1^D(N-1) \\ t_1^D(1) & t_1^D(2) & \ldots & t_1^D(N) \\ t_2^1(1) & t_2^1(2) & \ldots & t_2^1(N) \\ t_2^2(1) & t_2^2(2) & \ldots & t_2^2(N) \\ \vdots & \vdots & \ddots & \vdots \\ t_2^D(1) & t_2^D(2) & \ldots & t_2^D(N) \\ t_3^1(1) & t_3^1(2) & \ldots & t_3^1(N) \\ t_3^1(2) & t_3^1(3) & \ldots & t_3^1(N+1) \\ \vdots & \vdots & \ddots & \vdots \\ t_3^D(1) & t_3^D(2) & \ldots & t_3^D(N) \\ t_3^D(2) & t_3^D(3) & \ldots & t_3^D(N+1) \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \text{Detector 1} \\ \\ \\ \\ \text{Detector 2} \\ \\ \\ \\ \text{Detector 3} \\ \\ \end{matrix} \tag{24}$$

In the rest of this paper, we abbreviate the PPCA and KPPCA models working on data matrix $\widetilde{T}$ as **PPCA_3** and **KPPCA_3**.

PPCA and KPPCA models will impute all the missed entries in $\widetilde{T}$ that include those for *Detector 1* and *Detector 3*. Since a missed entry for *Detector 1* or *Detector 3* appears twice in $\widetilde{T}$, PPCA and KPPCA models may impute different values for these

entries. However, this will not cause a problem, since we are only interested in the imputing errors of *Detector 2*. These entries only appear once in $\tilde{T}$.

4. Test results

4.1. Testing dataset

In this section, we examine the imputing performance of different PPCA and KPPCA models. To give a fair comparison of the newly developed PPCA and KPPCA models, we choose the publicly available PeMS traffic datasets (PeMS) in the following test.

As pointed out in Qu et al. (2009) and Chen et al. (2012), the choice of aggregation time interval has little impact on the performance of PPCA based imputing methods. Hence, we only present the test results on one time aggregation scale, 5 min, in this paper.

As shown in many reports, traffic flow series in weekdays have different patterns than those in weekends and holidays. So, we only compare imputing performance of models on data of weekdays. The sampling period we used is between June 1, 2011 and August 31, 2011, in which weekends and holidays (e.g. July 4, 2011, the Independence Day) are excluded.

The specific dataset used in this paper was collected from Station-ID 402343 (*Detector 1*, the upstream one), 401670 (*Detector 2*, the middle one) and 400237 (*Detector 3*, the downstream one) which are located in S Valley Free Way in Santa Clara, CA, USA. There are four lanes at the studied road segment and we aggregate four data series into one data series. Since we cannot find a loop detector in PeMS datasets that always yields complete observations, we choose these detectors mainly because of their inherent small missing ratio.

In the PeMs project, the loop detectors send data every 30 s to the computer center. Thus, we will add up 10 basic observations to get one aggregated sample value. We assume an aggregated sample value is known (not missing/unknown), if we observe at least 5 basic observations in the corresponding 10 basic observations. Suppose the sum of all k observed basic values is φ, we will use $10\varphi/k$ as the aggregated value. After aggregation, the original missing ratios of three detector datasets are 2.17%, 0.99% and 1.04%, respectively. The missing ratios presented below do not include this original missing ratio.

It should be pointed out that we had tested these methods with data collected at other loop detectors in PeMS and the results are similar.

4.2. Test settings and evaluating indices

In the following tests, several missing data points are generated respectively with respect to the missing ratios and missing patterns. As discussed in Little and Rubin (1987) and Qu et al. (2009), we simulate three common missing patterns in the test: (1) missing completely at random (MCR), where the missing points are distributed independently of each other; (2) missing at random (MR), where the missing points are related to their neighboring points; and (3) the mixed MCR and MR missing pattern. We do not discuss the cases in which the missing data have certain particular patterns, since this is possibly related to a long term sensor malfunction.

For a given missing ratio, we generate 10 different samples of missing data, test all the imputing methods for all 10 samples and calculate the average imputing errors as the performance index. We use this method to abate the influence of sampling.

After the above PPCA and KPPCA models impute the missing values, the imputed values are compared with the true values to evaluate their performance. The closer the imputed values are to the true ones, the better the model is.

Similar to Qu et al. (2009), we had tested three popular error measures: Normalized Mean Absolute Error (NMAE), Normalized Root Mean Square Error (NRMSE), Root Mean Square Error (RMSE), to evaluate the distance between the imputed values and the true values. Results show that all models yield similar performance in these error criteria. Therefore, in the rest of this paper, only the Root Mean Square Error (RMSE) results will be given.

The Root Mean Square Error (RMSE) error is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{M}\sum_{m=1}^{M}\left(y_{real}^{(m)} - y_{impu}^{(m)}\right)^2} \tag{25}$$

where $y_{real}^{(m)}$ and $y_{impu}^{(m)}$ are the mth vectors of which the elements are the estimated values and the known real values, respectively. M denotes how many testing entries are used.

Generally, the best dimension of a PPCA/KPPCA model heavily depends on the similarity degree between traffic flow vectors in different days. However, there is no analytical formula to calculate the best dimension. In this paper, we use exhaustive search to get the best dimension. For a given scenario, we sequentially increase the dimension of latent variable space until the corresponding RMSE values level off. In order to avoid over-training, the optimal dimension of latent variable space is chosen as the first dimension value that reaches the saturated RMSE value. This trick is proven to be very fast and effective. Fig. 2 gives an example, in which the best dimensions are chosen as 20 for KPPCA_2 model and 21 for PPCA_2 model.

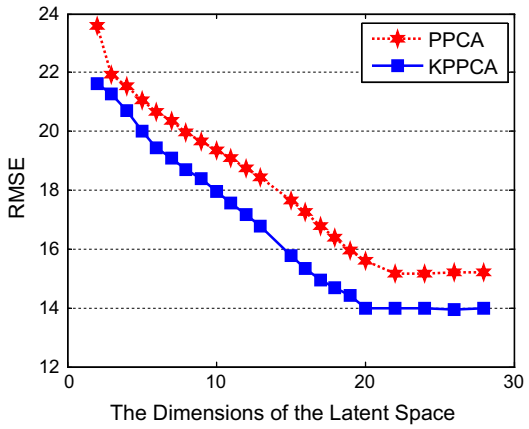

Fig. 2. The variation of reconstruction errors of **PPCA_2** and **KPPCA_2** with respect to the dimension of the latent variable space. Here, 1-month (July) data collected from *Detector 1, 2 and 3* are considered. The total missing ratio is 20% (that is, MCR points 10%, MR points 10%).

4.3. Comparison of imputing performance

To systematically compare the performance of PPCA and KPPCA based imputing methods, we designed four scenarios. In each scenario, we test their imputing errors under three different missing patterns with missing ratio varying from 5% to 30%. The missing ratio is defined as: the number of missing data points divides the number of total data points.

In agreement with the test results given in Qu et al. (2009), studies show that PPCA and KPPCA based imputing methods are quite stable for various missing ratios and missing point locations. Constrained by the length limit of this paper, we only present the test results for mixed MCR and MR missing pattern (the missing ratio for MCR and MR is kept as 50–50%) in the rest of this paper.

In the first scenario, we compare the performance of **PPCA_1** and **KPPCA_1** methods using 1-month (July) data collected from *Detector 2*. Fig. 3 shows the corresponding imputing errors given as RMSE values. Clearly, PPCA and KPPCA methods yield significantly smaller imputing errors than conventional simple average method (which fills a missing data point with the average of the known data collected at the same daily time point in the last 1 or 3 months), nearest historical imputing method (which fills a missing data point with the known data collected at the same daily time point in the nearest day), and spline interpolation method (which is to estimate the missing points with cubic spline interpolation (De Boor, 1978)). But the performance difference between PPCA and KPPCA methods are very small.

In the second scenario, we compare the performance of **PPCA_1** and **KPPCA_1** methods using 3-month (June to August) data collected from *Detector 2*. Fig. 4 shows the corresponding imputing errors given as RMSE values. We can see that the RMSE curves remained unchanged if being compared with Fig. 2. This indicates that the imputing errors cannot be reduced by simply feeding more data of one detector to PPCA and KPPCA models.

In the third scenario, we compare the performance of **PPCA_2** and **KPPCA_2** methods on 1-month (July) data collected from *Detector 1, 2 and 3*. Fig. 5 shows the corresponding imputing errors given as RMSE values. Apparently, **PPCA_2** and

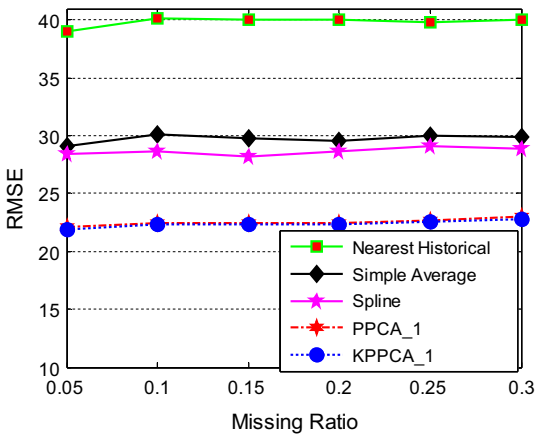

Fig. 3. Reconstruction errors of five imputation methods in the mixed MCR/MR missing pattern, when 1-month (July) data collected from *Detector 2* are considered.

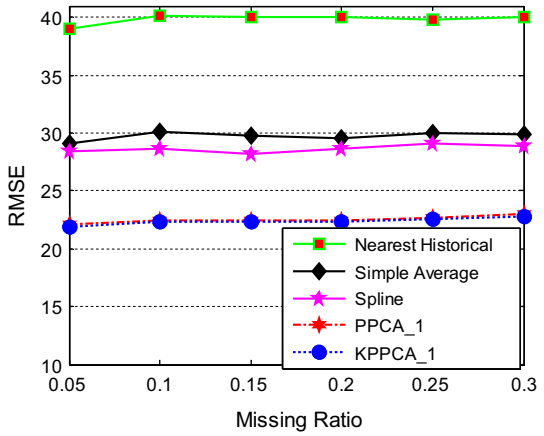

Fig. 4. Reconstruction errors of five imputation methods in the mixed MCR/MR missing pattern, when 3-month (June to August) data collected from *Detector 2* are considered.

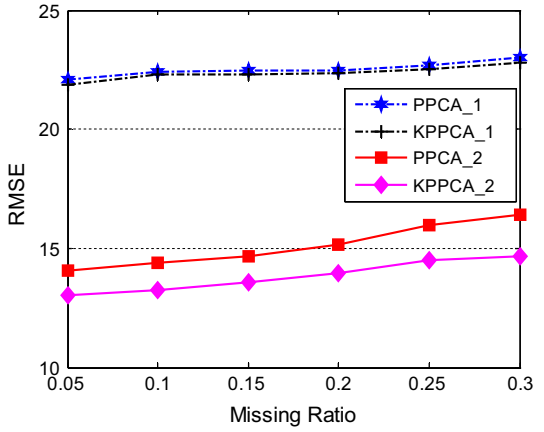

Fig. 5. Reconstruction errors of **PPCA_2** and **KPPCA_2** methods in the mixed MCR/MR missing pattern, when 1-month (July) data collected from *Detector 1, 2 and 3* are considered. Only the imputing data for *Detector 2* are used to calculate the reconstruction errors. No time lag is considered.

KPPCA_2 methods yield notably smaller imputing errors than **PPCA_1** and **KPPCA_1** methods. This plainly demonstrates the benefits of considering spatial dependences in PPCA and KPPCA methods. Moreover, KPPCA method gives even smaller imputing errors since the spatial dependences are nonlinear.

In the fourth scenario, we compare the performance of **PPCA_3** and **KPPCA_3** methods on 1-month (July) data collected from *Detector 1, 2 and 3*, when the influence of time lag is considered. Fig. 6 shows the corresponding imputing errors given as RMSE values. Apparently, **KPPCA_3** method produces even smaller imputing errors than **PPCA_2** and **KPPCA_2** methods. However, **PPCA_3** produces significantly larger imputing errors than **PPCA_2** method.

The key change here comes from the increase of the row size for the data matrix. For KPPCA model, this increases the sample size of the data and allows us a chance to better recover the nonlinear structure hidden in the data. Fig. 6 proves that KPPCA method considering both temporal and spatial dependences would outperform PPCA and KPPCA methods only considering spatial dependences.

For PPCA model, the introduction of time lag does not increase the sample size of the data but instead increases the feature size of the data. As discussed in Hoyle (2008), when the feature size is relatively large and the sample size is relatively small, the EM algorithm may fail to find an appropriate parameter sets for the model. Thus, feeding too much data into PPCA model may lead to biased estimation and thus degrade imputing results.

For the middle detector, the distances to the upstream/downstream detectors are 1.78 km and 0.63 km in our studies. However, the distances of neighboring detectors are not the major factors here, because the time lag must be an integer multiple of the size of aggregation time window, no matter where the locations of upstream/downstream detectors are. We had tested other selections of time lag, too. Results indicated that a time lag 1 is the best choice in this particular case.

Moreover, we examine empirical distributions for the deviations time series of original/reconstructed traffic flows. Here, the deviations are calculated as the differences between original/reconstructed traffic flow and the simple average intra-day

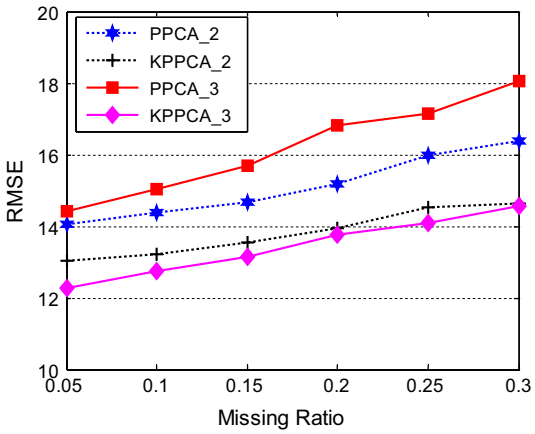

Fig. 6. Reconstruction errors of **PPCA_3** and **KPPCA_3** methods in the mixed MCR/MR missing pattern, when 1-month (July) data collected from *Detector 1, 2 and 3* are considered. Only the imputing data for *Detector 2* are used to calculate the reconstruction errors. Time lag is appropriately considered as Eq. (22).

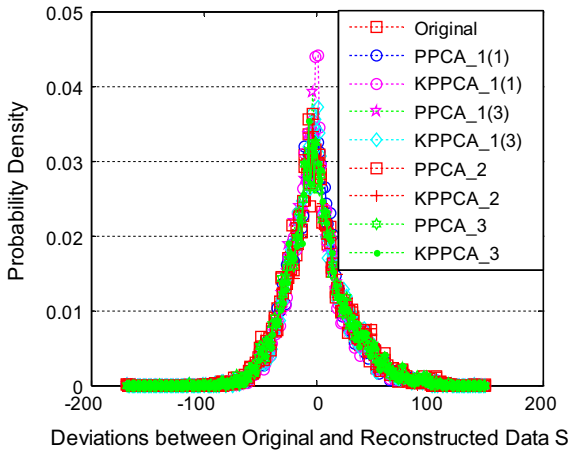

Fig. 7. Distribution of the deviations between the intra-day trend and reconstructed daily traffic flow by different imputing methods. The total missing ratio for mixed MCR/MR pattern is 30%.

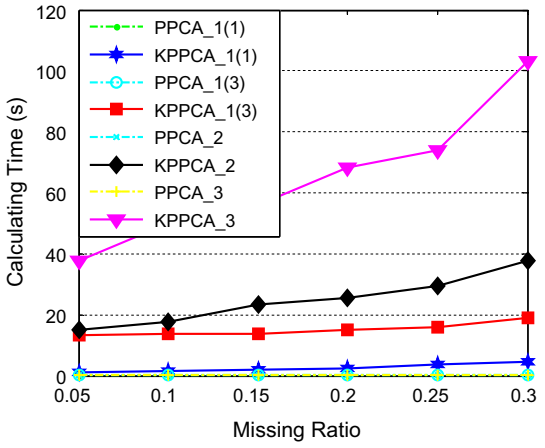

Fig. 8. Calculating time of PPCA and KPPCA methods in the mixed MCR/MR missing pattern under different missing ratios. The testing platform has the following settings: CPU:Intel(R) Core(TM) i7-3770 3.40 GHz; RAM: 8G; Windows: Windows 7 (64 bits).

trend; see Qu et al. (2009). If the histograms for the deviations of the original and reconstructed traffic flows are identical, the reconstructed traffic flow data will preserve the important statistical feature of the data. Otherwise, imputation may pollute the original traffic flow data. Fig. 7 verifies that all PPCA based imputing methods nicely retain the statistical property of original data.

In addition, we also compare the calculation time consumed by different PPCA and KPPCA methods, since running speed is an important factor when we apply imputing methods in online systems. As shown in Fig. 8, the calculation times for PPCA methods are negligible, if be compared with thost for KPPCA methods. Even the most time consuming **PPCA_3** method can finish imputing task in 1 s. Thus, we strongly recommand **PPCA_2** method rather than all the other PPCA and KPPCA methods in online systems (i.e. online prediction systems (Chen et al., 2012)), since it gives relatively small imputing errors and fast running speed. On the other side, we highly recommand **KPPCA_3** method for off-line systems which do not care much about the imputing time cost, since it gives the smallest imputing errors.

5. Conclusions

Rapid advances in information technologies are enabling us to collect far more information of transportation systems at all levels. However, the missing data problem still hinders us in many applications. Aiming to solve this problem, we discussed whether we could promote missing data imputing performance by fusing the information of multiple points in this paper.

Results showed that using spatial and temporal dependence could help reduce imputing errors significantly for PPCA and KPPCA methods. Moreover, such dependence is naturally nonlinear and would be better utilized via KPPCA based imputing method rather than PPCA based method. This indicates that we could further released the power of existing traffic flow prediction/imputing methods, if we can appropriately incorporate the information of neighboring detecting points. However, the calculation time of KPPCA is significantly longer than PPCA. Hence, PPCA method considering only spatial dependence is still highly recommended for online systems.

Kernel method is very popular in machine learning areas nowadays. It is one of the most convenient techniques to introduce nonlinear mapping relations into data structure. However, we do not claim that kernel method is the only tool to obtain such impressive improvements in missing data imputing. Our purpose here is to draw researchers' attentions to further investigate the merits of nonlinear modeling for traffic flows.

The proposed method does not consider outlier detection in this paper. As a result, it may not distinguish some temporarily (often meanwhile spatially isolated) patterns (which are usually caused special events) from long-term patterns in the traffic state. The imputing performance might be degraded sometimes, since KPPCA model possibly makes overtraining to "learn" those abnormal patterns. One remedy is to simply apply Robust PCA method and some alternative methods to filter out the abnormal pattern, before we carry out imputing tasks. Please check Jin et al. (2008) and Qu et al. (2009) for detailed discussions.

Although the obtained results are very promising, it should be pointed out that the best way of describing traffic spatial features is still an open problem (Karlaftis, 2012). Although the proposed column operation skill is efficient in PPCA/KPPCA based traffic flow analysis, it cannot always be directly applied into many other popular methods. The complicated spatial–temporal correlations existing in urban road network also need far more interpretations. We would like to discuss other dependence modeling skills for prediction/imputing models in our coming reports.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China 51278280, National Basic Research Program of China (973 Project) 2012CB725405, Hi-Tech Research and Development Program of China (863 Project) 2011AA110301.

References

Ahmed, M.S., Cook, A.R., 1979. Analysis of freeway traffic time-series data by using Box-Jenkins techniques. Transportation Research Record 722, 1–9.
Bickel, P.J., Chen, C., Kwon, J., Rice, J., van Zwet, E., Varaiya, P., 2007. Measuring traffic. Statistical Science 22 (4), 587–597.
Boyd, S., Vandenberghe, L., 2004. Convex Optimization. Cambridge University Press.
Chen, C., Kwon, J., Rice, J., Skabardonis, A., Varaiya, P., 2002. Detecting errors and imputing missing data for single loop surveillance systems. Transportation Research Record 1855, 160–167.
Chen, C., Wang, Y., Li, L., Hu, J., Zhang, Z., 2012. The retrieval of intra-day trend and its influence on traffic prediction. Transportation Research Part C: Emerging Technologies 22, 103–118.
De Boor, C., 1978. A Practical Guide to Splines. Springer-Verlag.
Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B 39 (1), 1–38.
Der Voort, M., Dougherty, M., Watson, S., 1996. Combining Kohonen maps with ARIMA time series models to forecast traffic flow. Transportation Research Part C: Emerging Technologies 4 (5), 307–318.
Haworth, J., Cheng, T., 2012. Non-parametric regression for space–time forecasting under missing data. Computers, Environment and Urban Systems 36 (6), 538–550.
Hoyle, D.C., 2008. Automatic PCA dimension selection for high dimensional data and small sample sizes. Journal of Machine Learning Research 9, 2733–2759.

Ilin, A., Raiko, T., 2010. Practical approaches to principal component analysis in the presence of missing values. Journal of Machine Learning Research 11, 1957–2000.
Jin, X., Zhang, Y., Li, L., Hu, J., 2008. Robust PCA based abnormal traffic flow pattern isolation and loop detector fault detection. Tsinghua Science and Technology 13 (6), 829–835.
Kamarianakis, Y., Prastacos, P., 2003. Forecasting traffic flow conditions in an urban network: comparison of multivariate and univariate approaches. Transportation Research Record 1857, 74–84.
Kamarianakis, Y., Shen, W., Wynter, L., 2012. Real-time road traffic forecasting using regime-switching space-time models and adaptive LASSO. Applied Stochastic Models in Business and Industry 28 (4), 297–315.
Karlaftis, M.G., 2012. Discussion. Applied Stochastic Models in Business and Industry 28 (4), 316–318.
Karlaftis, M.G., Vlahogianni, E.I., 2011. Statistical methods versus neural networks in transportation research: differences, similarities and some insights. Transportation Research Part C: Emerging Technologies 19 (3), 387–399.
Lawrence, N., 2005. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. Journal of Machine Learning Research 6, 1783–1816.
Li, Y., Li, Z., Li, L., Zhang, Y., 2013. Comparison on PPCA, KPPCA and MPPCA based missing data imputing for traffic flow. In: Proceedings of International Conference on Transportation Information and Safety (in press).
Little, R.J.A., Rubin, D.B., 1987. Statistical Analysis with Missing Data. John Wiley & Sons Press.
McLachlan, G., Krishnan, T., 2008. The EM Algorithm and Extensions, second ed. Wiley, Hoboken, NJ, USA.
Min, W., Wynter, L., 2011. Real-time road traffic prediction with spatio-temporal correlations. Transportation Research Part C: Emerging Technologies 19 (4), 606–616.
Min, X., Hu, J., Zhang, Z., 2010. Urban traffic network modeling and short-term traffic flow forecasting based on GSTARIMA model. In: Proceedings of IEEE Conference on Intelligent Transportation Systems, pp. 1535–1540.
Moller, M.F., 1993. A scaled conjugate gradient algorithm for fast supervised learning. Neural Networks 6 (4), 525–533.
Ni, D., Leonard II, J.D., 2005. Markov Chain Monte Carlo multiple imputation using Bayesian Networks for incomplete intelligent transportation systems data. Transportation Research Record 1935, 57–67.
Ni, D., Leonard II, J.D., Guin, A., Feng, C., 2005. Multiple imputation scheme for overcoming the missing values and variability issues in ITS data. ASCE Journal of Transportation Engineering 131 (12), 931–938.
PeMS, California Performance Measurement System. <http://pems.eecs.berkeley.edu>.
Qu, L., Li, L., Zhang, Y., Hu, J., 2009. PPCA-based missing data imputation for traffic flow volume: a systematical approach. IEEE Transactions on Intelligent Transportation Systems 10 (3), 512–522.
Ramezani, A., Moshiri, B., Abdulhai, B., Kian, A.R., 2010. Distributed maximum likelihood estimation for flow and speed density prediction in distributed traffic detectors with Gaussian mixture model assumption. IET Intelligent Transport Systems 6 (2), 215–222.
Roweis, S., Ghahramani, Z., 1999. A unifying review of linear Gaussian models. Neural Computation 11 (2), 305–345.
Sanguinetti, G., Lawrence, N.D., 2006. Missing data in kernel PCA. Lecture Notes in Computer Science 4212, 751–758.
Schölkopf, B., Smola, A., Müller, K.-R., 1997. Kernel principal component analysis. Lecture Notes in Computer Science 1327, 583–588.
Sharma, S., Lingras, P., Zhong, M., 2004. Effect of missing values estimations on traffic parameters. Transportation Planning and Technology 27 (2), 119–144.
Shawe-Taylor, Cristianini, J.N., 2004. Kernel Methods for Pattern Analysis. Cambridge University Press.
Smith, B.L., Scherer, W.T., Conklin, J.H., 2003. Exploring imputation techniques for missing data in transportation management systems. Transportation Research Record 1836, 132–142.
Stathopoulos, A., Karlaftis, M.G., 2003. A multivariate state space approach for urban traffic flow modeling and prediction. Transportation Research Part C: Emerging Technologies 11 (2), 121–135.
Stathopoulos, A., Tsekeris, T., 2006. Methodology for processing archived ITS data for reliability analysis in urban networks. IEE Proceedings of Intelligent Transportation Systems 153 (1), 105–112.
Sun, S., Xu, M., 2011. Variational inference for infinite mixtures of Gaussian processes with applications to traffic flow prediction. IEEE Transactions on Intelligent Transportation Systems 12 (2), 466–475.
Sun, S., Zhang, C., Yu, G., 2006. A Bayesian network approach to traffic flow forecasting. IEEE Transactions on Intelligent Transportation Systems 7 (1), 124–132.
Sun, S., Huang, R., Gao, Y., 2012. Network-Scale Traffic Modeling and Forecasting with Graphical Lasso and Neural Networks. Journal of Transportation Engineering 138 (11), 1358–1367.
Tan, H., Feng, G., Feng, J., Wang, W., Zhang, Y.-J., Li, F., 2013. A tensor-based method for missing traffic data completion. Transportation Research Part C: Emerging Technologies 28, 15–27.
Tipping, M.E., Bishop, C.M., 1999. Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B 21 (3), 611–622.
Turner, S., Albert, L., Gajewski, B., Eisele, W., 2000. Archived intelligent transportation system data quality: preliminary analyses of San Antonio TransGuide data. Transportation Research Record 1719, 77–84.
van Lint, J.W.C., Hoogendoorn, S.P., van Zuylen, H.J., 2005. Accurate freeway travel time prediction with state-space neural networks under missing data. Transportation Research Part C: Emerging Technologies 13 (5–6), 347–369.
Vlahogianni, E.I., Golias, J.C., Karlaftis, M.G., 2004. Short-term traffic forecasting: overview of objectives and methods. Transportation Reviews 24 (5), 533–557.
Vlahogianni, E.I., Karlaftis, M.G., Golias, J.C., 2005. Optimized and meta-optimized neural networks for short-term traffic flow prediction: a genetic approach. Transportation Research Part C: Emerging Technologies 13 (3), 211–234.
Williams, B.M., 2001. Multivariate vehicular traffic flow prediction evaluation of ARIMAX modeling. Transportation Research Record 1776, 194–200.
Yin, W., Murray-Tuite, P., Rakha, H., 2012. Imputing erroneous data of single-station Loop detectors for nonincident conditions: comparison between temporal and spatial methods. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations 16 (3), 159–176.
Zhang, J., Wang, F.-Y., Wang, K., Lin, W.-H., Xu, X., Chen, C., 2011. Data-driven intelligent transportation systems: a survey. IEEE Transactions on Intelligent Transportation Systems 12 (4), 1624–1639.
Zhong, M., Lingras, P., Sharma, S., 2004a. Estimation of missing traffic counts using factor, genetic, neural, and regression techniques. Transportation Research Part C: Emerging Technologies 12 (2), 139–166.
Zhong, M., Sharma, S., Lingras, P., 2004b. Genetically designed models for accurate imputations of missing traffic counts. Transportation Research Record 1879, 71–79.
Zhong, M., Sharma, S., Liu, Z., 2005. Assessing robustness of imputation models based on data from different jurisdictions: examples of Alberta and Saskatchewan, Canada. Transportation Research Record 1917, 116–126.