Hindawi
Journal of Advanced Transportation
Volume 2017, Article ID 4018409, 13 pages
https://doi.org/10.1155/2017/4018409

Research Article

A Streaming Algorithm for Online Estimation of Temporal and Spatial Extent of Delays

Kittipong Hiriotappa, Suttipong Thajchayapong, Pimwadee Chaovalit, and Suporn Pongnumkul

National Electronics and Computer Technology Center (NECTEC), 112 Thailand Science Park, Phahonyothin Road, Khlong Nueng, Khlong Luang, Pathum Thani 12120, Thailand

Correspondence should be addressed to Suttipong Thajchayapong; suttipong.thajchayapong@nectec.or.th

Received 2 July 2016; Accepted 20 September 2016; Published 10 January 2017

Academic Editor: David F. Llorca

Knowing traffic congestion and its impact on travel time in advance is vital for proactive travel planning as well as advanced traffic management. This paper proposes a streaming algorithm to estimate temporal and spatial extent of delays online which can be deployed with roadside sensors. First, the proposed algorithm uses streaming input from individual sensors to detect a deviation from normal traffic patterns, referred to as *anomalies*, which is used as an early indication of delay occurrence. Then, a group of consecutive sensors that detect anomalies are used to temporally and spatially estimate extent of delay associated with the detected anomalies. Performance evaluations are conducted using a real-world data set collected by roadside sensors in Bangkok, Thailand, and the NGSIM data set collected in California, USA. Using NGSIM data, it is shown qualitatively that the proposed algorithm can detect consecutive occurrences of shockwaves and estimate their associated delays. Then, using a data set from Thailand, it is shown quantitatively that the proposed algorithm can detect and estimate delays associated with both recurring congestion and incident-induced nonrecurring congestion. The proposed algorithm also outperforms the previously proposed streaming algorithm.

1. Introduction

Traffic congestion has become a major problem in big cities around the world, causing economic, environmental, and health issues. Existing solutions for addressing the traffic congestion problem are to increase road capacity, to use alternative transportation mode (e.g., mass transit systems), to restrict usage on particular roads at particular times (e.g., congestion charge), and to use existing roads more efficiently [1]. While the first three solutions require extensive building of new infrastructure and/or studying of public opinions, the last solution can be more promptly implemented as it mainly involves utilizing traffic information in a more efficient manner.

Intelligent Transport Systems (ITS) have provided efficient means of gathering, processing, and disseminating relevant traffic information by utilizing the capabilities of sensor networks on roadside and onboard vehicles [2]. One of the most effective and practical ways is to promptly provide relevant traffic information in advance to enable drivers to avoid congested routes and traffic management centers to proactively manage traffic flow. This would enable not only drivers to estimate their travel times, but also traffic management centers to estimate the severity of congestion and initiate appropriate strategies to return traffic to normal.

Even though numerous traffic estimation and prediction methods have been proposed in the past decades, recent literature review in [3] indicates that there are still ten challenges that need research attentions. One of these challenges is on how to use short-term traffic estimation to reflect a macroscopic point of view of traffic condition, which is very essential for traffic management strategies. While most short-term traffic estimation methods focus on estimating flow, travel time, or traffic condition itself, few recent studies have shifted to incorporating new information that reflects further insights on future traffic condition. These new information include travel time variability which reflects reliability [4] and delay associated with anomalies which reflects severity [5–7].

The focus of this paper is on detecting anomalies and estimating delay on a freeway segment. Anomaly detection and delay estimation are essential for both drivers and traffic management centers as they reflect severity of traffic congestion, which in turn can be used to minimize its impact (e.g., by alerting drivers to change route or sending in a tow truck to remove a disabled vehicle). While anomaly detection has received more attention in the past decade [8], delay estimation has not equally received the same attention. As detailed in Section 2 below, the majority of previous studies have focused on estimating and predicting travel time under normal traffic condition [9–15], while relatively fewer methods focus particularly on traffic delay [5–7, 10, 16]. Furthermore, the methods proposed in [5, 7] are not designed for real-time operation, while the study in [6] focuses only on traffic delay in work zones. One of the most recent studies in [16] provides insight on the capabilities of existing models but the focus is on signalized intersections.

The contributions of this paper can be summarized as follows. First, the proposed algorithm is designed to operate online and can be deployed with existing roadside sensors. Second, it can detect anomalies and achieve higher detection rates and lower false positive rates (i.e., false alarm rates) compared to an existing approach in [17]. Third, the algorithm can estimate delay by first estimating the spatial extent of delay using a group of roadside sensors and then temporally estimate delay using empirically derived equations. Finally, for practical purposes, the benefit of selecting suitable input time window size and prediction horizon is discussed.

This paper proposes an algorithm based on a new framework which includes both anomaly detection and delay estimation so it is a significantly enhanced version of the algorithm in [18], which focuses only on detecting incidents. First, the spatial and temporal estimation of delay is incorporated (explained in detail in Section 4). Second, it is shown that the enhanced algorithm in this paper can be used to detect the presence of shockwaves and estimate their associated delays (explained in Section 5). Thirdly, the real-world data sets used to assess the proposed algorithm in this paper are much more significant than the one used in [18] with 44 more delay cases with additional six months of data collection (explained in Section 5). Finally, to show the potential of the proposed algorithm being deployed in a different location, the well-known NGSIM data [19] is also used in the assessment, which has not been done in [18].

The rest of this paper is organized as follows. Related work is discussed in Section 2. Research framework is presented in Section 3. Then, Section 4 describes the proposed algorithm in detail. Performance evaluations using real-world data are described in Section 5, and the results are discussed in Section 6. Finally, Section 7 concludes this paper and discusses future work.

2. Related Work

The majority of previous studies have focused on estimating and predicting travel time under normal traffic conditions [9–15]. Among them, the most widely used technologies are probe vehicle and automatic vehicle identification (AVI). Probe vehicle technologies are installed in the vehicles themselves to communicate their locations to some remote servers where their travel times are calculated [10]. One of the most well-known probe vehicle technologies is the Global Positioning System (GPS). Unlike probe vehicles, AVI technologies are designed to be installed on roadside to identify individual vehicles in different locations [9]. Using AVI, the travel time of each vehicle can be calculated as the difference between the times that the vehicle passes two consecutive locations [11]. Well-known AVI technologies are radio frequency identification (RFID) and license plate recognition. Algorithms used to estimate travel times are support vector regression models [9], a three-layer neural network model [10], a low-pass adaptive filtering algorithm [11], among others. Even though using travel time provides the most accurate and convenient way to estimate and predict delay, it requires an adequate number of sensors installed in vehicles and/or on roadways which are not yet widely available in many areas of developing countries.

Besides using probe vehicles and AVI to directly estimate travel time, a group of indirect methods have also been proposed where travel time is estimated and predicted from traffic flow [12–15]. They use similar assumptions that there are certain degrees of relationship between flow and delay. Commonly employed techniques are regression, artificial intelligence, and statistical models derived from historical data. In particular, it is shown that travel time can be accurately predicted up to 20 minutes in advance using linear regression [12–14]. Even though these methods have been shown to perform well under normal traffic conditions, they need adequate historical data which may not always be available on every road segment. Also, transient changes under congested traffic condition may have an inconsistent impact on the estimation accuracy [15].

Relatively few methods have been designed to estimate traffic delay, that is, the additional travel time that occurs under congested conditions. The methods in this group rely on a primary assumption that speed does not change significantly over a section of roadway. An approach proposed in [20] follows this assumption when incident-induced delays can be estimated by first establishing reference incident-free travel profiles. Based on reference profiles, one can estimate incident-induced delays more accurately. Another work [21] estimates incident delay by deploying both incident duration model and reduced capacity model into the deterministic queuing model, thereby introducing incident-induced traffic delays of stochastic nature into an otherwise deterministic traffic. This same principle applies to calculating traffic delays caused by precipitation where archived weather and traffic data are used in traffic delay estimation [22]. Another good example is the method proposed in [5] which estimates delay based on speed measurements. However, the method in [5] can only work offline and requires the knowledge of the location of an accident. The aim of the method proposed in this paper is similar to the aim in [5], but the main difference is that our method is designed primarily to work online, that is, to process real-time speed inputs in a streaming manner.

In our previous works, we have proposed algorithms to detect and classify different types of traffic anomalies [8, 23], to transfer the detection capability to roadside sensors at a different location [24], and recently to infer traffic transition at locations where local traffic data is not available [25]. The algorithm proposed in this paper focuses on anomalies that cause delay so it uses a different approach from our previous works in [8, 23]. More importantly, the proposed algorithm is a significant extension of our previous work in [18] where the capabilities to detect anomalies at individual sensors, to detect shockwaves, and to estimate traffic delay are shown.

3. Research Framework

In this section, we present research framework including definitions of certain terminologies that are essential for developing the proposed algorithm in Section 4. Firstly, it is important to define *temporal and spatial extent of delay*. *Delay* is defined as an additional travel time experienced by vehicles due to circumstances that impede the movement of traffic [27]. *Spatial extent* refers to an area on the road segment where the vehicles are expected to experience delay. *Temporal extent* in this paper refers to short-term estimation; that is, it indicates the delay vehicles are expected to experience in the near future. *Short-term* is defined as a time period from a few seconds to a few hours into the future [3]. For example, for the expressway we analyzed in this paper, short-term refers to the period of $(0, 15]$ minutes.

Delay occurs after traffic condition changes from *normal* to *disruptive* condition [28]. *Normal* condition refers to the scenario when incoming traffic flow is well below the freeway segment capacity. *Disruptive* condition can be associated with either *recurring* or *nonrecurring* congestion. *Recurring* congestion refers to the scenario when traffic flow exceeds freeway capacity (e.g., during morning rush hours on weekdays). *Nonrecurring* congestion refers to the scenarios where the capacity of the freeway segment is reduced by unexpected events (e.g., accidents and disabled vehicles). An *anomaly* in this paper refers to a deviation from expected traffic patterns which can occur in recurring congestion (e.g., speed drops due to shockwave) [25] or precede nonrecurring congestion (e.g., speed drops due to accidents and disabled vehicles) [8].

Figure 1 shows how the proposed algorithm can be deployed with roadside sensors [2] on a freeway segment. A typical roadside sensors system shown in Figure 1 consists of sensors placed consecutively on a freeway segment. In each sensor, there are two main units. A sensor unit is responsible for collecting traffic data using sensing technologies such as magnetometers, radars, lasers, and video cameras [29]. A communication unit is tasked with sending traffic data to a data collection module using wireless communication technologies such as GPRS, 3G, or Bluetooth depending on the sensors' topology and which technologies are available locally. Then, the collected traffic data are fed to the algorithm module where they are processed by certain algorithms for particular applications. The proposed algorithm in this paper is deployed in the algorithm module where it detects anomalies and estimates delay. The delay information provided by the proposed algorithm can then be used for particular applications including advanced traveler information systems and advanced traffic management systems.

4. The Proposed Algorithm

In this section, the proposed algorithm is presented. First, an overview is given in Section 4.1 on how each operation of the proposed algorithm processes the input traffic data and interacts with one another. Then, Section 4.2 explains two important input parameters which are used to adjust the proposed algorithm. Section 4.3 presents the anomaly detection operation which is used to detect the occurrence of delay. Finally, Section 4.4 describes how the proposed algorithm estimates delay spatially and temporally.

4.1. Overview

As shown in Figure 2, the proposed algorithm operates online in a streaming manner and uses temporal samples of traffic data obtained from roadside sensors as input. For a given time step n and an input time window size L, each operation is performed with L samples of input $V_{m,n} = [v_{m,n}, v_{m,n-1}, \ldots, v_{m,n-L+1}]$ from each roadside sensor, where $v_{m,n}$ refers to the average speed of the vehicles that pass through sensor m at time step n. The proposed algorithm consists of two main operations, namely, anomaly detection (block II in Figure 2) and delay estimation (blocks III and IV in Figure 2). The aim of anomaly detection is to detect deviations from normal traffic patterns that cause traffic delay. Once an anomaly is detected, the algorithm activates delay estimation operation where the aim is to estimate an additional travel time vehicles are expected to experience on a given road segment.

4.2. Definitions of Input Time Window Size (L) and Prediction Horizon (P)

Before we describe further how the proposed algorithm performs anomaly detection and delay estimation, it is important to define time window size L and prediction horizon P, which are two important input parameters as shown in Figure 2. The input time window size L defines the number of temporal samples of input (in unit of time) the proposed algorithm can use to detect anomaly and estimate delay. The prediction horizon P defines how long into the future (also in unit of time) the vehicles would experience the delay estimated by proposed algorithm. In other words, when a delay of d is estimated by the algorithm at time n, the freeway segment is expected the experience similar delay d during the period $[n, n+P]$. Even though it is well known that increasing P would reduce estimation accuracy, this paper assesses the degree of change of the estimation accuracy in respect to changes in the values of P.

4.3. Anomaly Detection

The proposed algorithm is developed based on the assumption that traffic delay is caused by anomalies so delay can be promptly identified by first detecting anomalies. Based on our experiments as well as the previous findings in [30], speed, flow, and occupancy are suitable for identifying traffic anomalies so they are selected as inputs. As real-world data may be incomplete, inconsistent, and noisy, the first step is to perform data cleansing as shown in block I in Figure 2. For the proposed algorithm, an

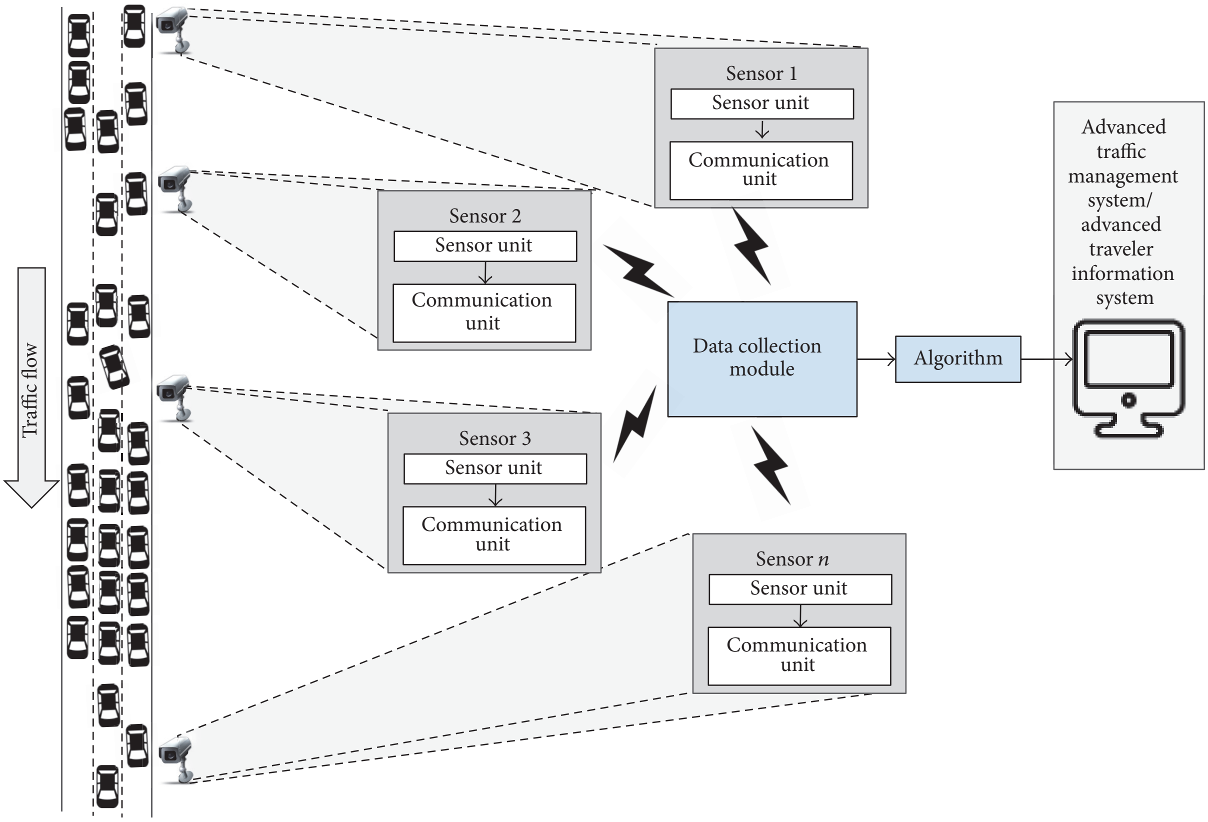


Figure 1: How the proposed algorithm can be deployed with roadside sensors on a freeway segment. The proposed algorithm is the algorithm module which uses traffic data collected from sensors to detect anomalies and estimate delay.

exponential moving average derived from DELOS smoothing [31] is used to interpolate missing data and remove noise. The exponential moving average is chosen particularly to give more weights to the latest speed samples as they are more associated with delay occurrence.

The anomaly detection operation is derived based on the sliding window approach in [32] but modified to assess each input time window individually whether it is associated with normal or anomalous traffic patterns. The proposed algorithm uses pattern matching, where each real-time input pattern consisting of L samples of traffic data $V_{m,n} = [v_{m,n}, v_{m,n-1}, \ldots, v_{m,n-L+1}]$ from roadside sensors. The input $V_{m,n}$ is identified as normal or anomalous by assessing its similarity with a set of basis patterns $U_{m,n} = [u_{m,n}, u_{m,n-1}, \ldots, u_{m,n-L+1}]$ consisting of normal patterns U_m^N and anomalous patterns U_m^A. An anomaly is detected if the input pattern $V_{m,n}$ is more similar to U_m^A than to U_m^N. In practice, $V_{m,n}$ and $U_{m,n}$ can be speed, flow, and occupancy measured from both normal and anomalous conditions a priori and stored in a database.

The main challenge for anomaly detection is that anomalies can cause speed, flow, and occupancy patterns to behave differently in time, space, and shape. For example, it is observed that a disruption occurring between a pair of upstream and downstream roadside sensors causes speed to drop at the upstream sensor while remain constant at a downstream sensor. On the other hand, the same disruption may cause flow to drop at both upstream and downstream sensors. Furthermore, the same disruption can cause the speed patterns to drop within seconds under a scenario where vehicles are moving at relatively high speeds. In contrast, it may take place in several minutes in another scenario depending on traffic flow and drivers' behaviors. Also, the shape of speed drop patterns may be linear (with different slopes), exponential, logarithmic, or even sigmoid as found in [24]. To address these challenges, Dynamic Time Warping is used in the proposed algorithm.

Dynamic Time Warping (DTW) is a pattern similarity measurement between time series proposed in [33] to overcome limitations of traditional Euclidean distance by introducing warping axis to adjust the error measurement. Figure 3 shows distance measurement between two time series patterns V and U using DTW. It can be seen in Figure 3(a) that at first V and U do not appear to be similar by using Euclidean distance measurement to compare points at the same time. However, as shown in Figures 3(b) and 3(c), DTW offers a more flexible and efficient approach where similarity between two time series patterns can be assessed by comparing points at different times. This is equivalent to "wrapping" the time series before comparing them.

In practice, the DTW would operate in an algorithm module in Figure 1 to assess similarity between an input pattern V measured by a roadside sensor and individual basis pattern U. Given an input time window size L, a computer programming language is used to implement DTW

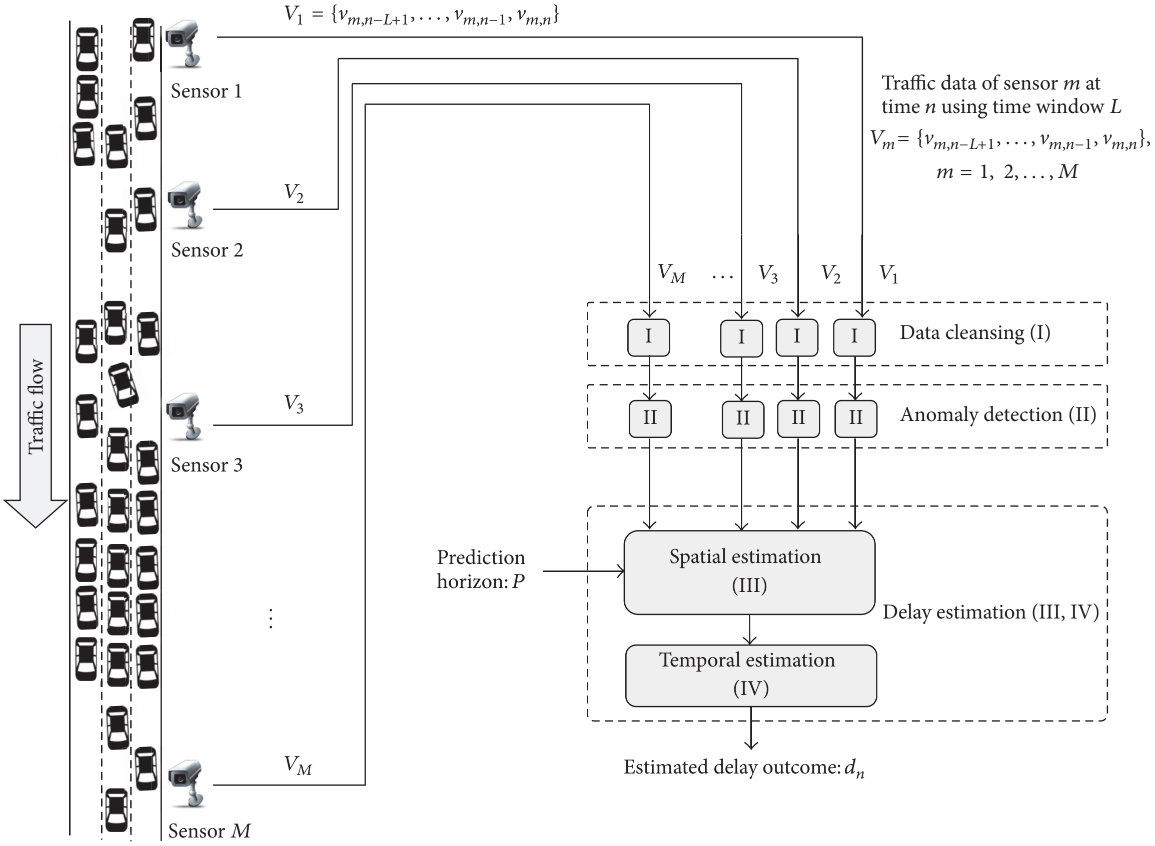
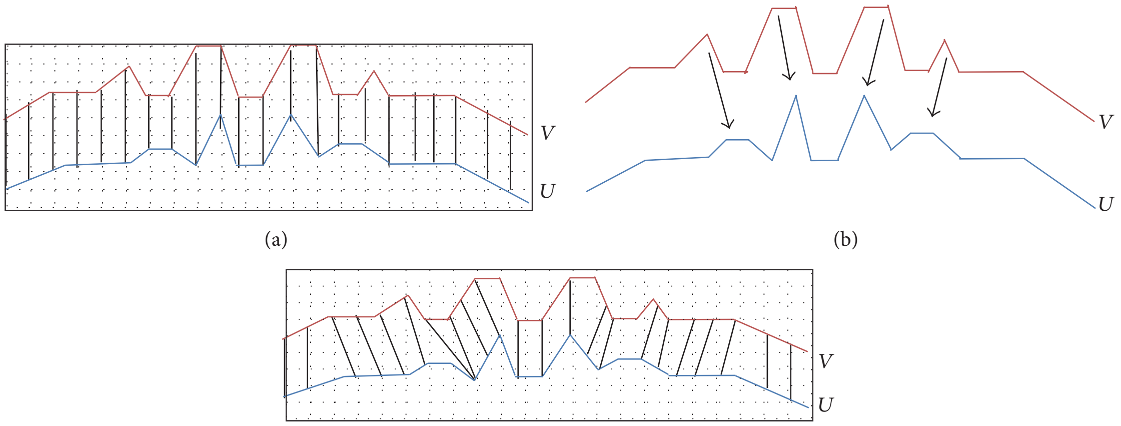

$V_1 = \{v_{m,n-L+1}, \ldots, v_{m,n-1}, v_{m,n}\}$
Sensor 1
Sensor 2
Sensor 3
Sensor M
V_2
V_3
V_M
Traffic flow
Traffic data of sensor m at time n using time window L
$V_m = \{v_{m,n-L+1}, \ldots, v_{m,n-1}, v_{m,n}\}$, $m = 1, 2, \ldots, M$
V_M … V_3 V_2 V_1
I
Data cleansing (I)
II
Anomaly detection (II)
Prediction horizon: P
Spatial estimation (III)
Temporal estimation (IV)
Delay estimation (III, IV)
Estimated delay outcome: d_n

Figure 2: Overview of the proposed algorithm.



(a)

(b)

(c)

Figure 3: An example of distance measurements between two time series using Dynamic Time Warping [26]. (a) Euclidean distance measurement between two time series. (b) DTW attempts to warp in time axis to find the correct measurement. (c) Distance measurement using DTW.

by constructing a L-by-L matrix where the number of rows and columns are equivalent to the length of V and U, respectively. Each matrix element (i, j) corresponds to the alignment between points v_i and u_j. Then, a warping path $W = w_1, w_2, \ldots, w_K$; $L \leq K < 2L - 1$ is constructed as a set of matrix elements that defines a point mapping in order to measure the distance between V and U. Similarity between V and U can be calculated based on these warping paths.

Based on DTW, the proposed algorithm is flexible enough to cope with these pattern differences and subsequently compare V and U more efficiently. If the input pattern V is found to be similar to the basis pattern U associated with anomalies, the proposed algorithm declares that an anomaly

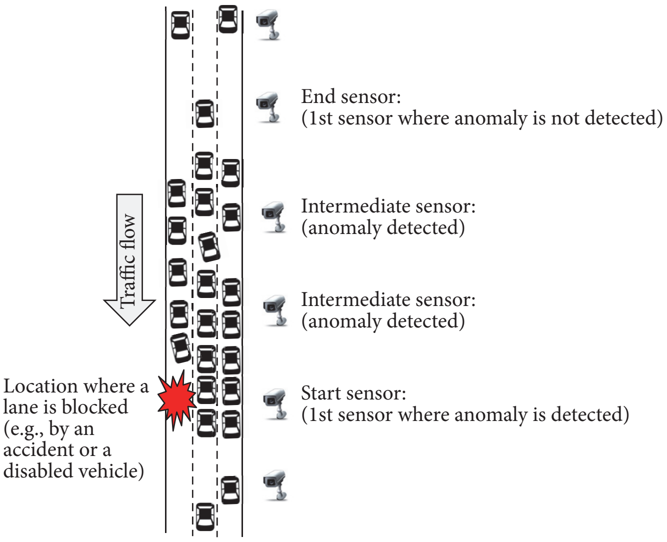


Figure 4: An illustration of how the proposed algorithm finds start, intermediate, and end sensors from speed patterns measured by roadside sensors. Start sensor is the first downstream sensor that detects an anomaly. End sensor is the first upstream sensor that an anomaly is not detected. Intermediate sensors are used to confirm that the congested area has extended upstream from start sensor to end sensor.

is detected and proceeds to delay estimation. We note that anomalies can also be detected by multiple sensors which can be used in the delay estimation operation.

4.4. Delay Estimation. Delay estimation operation consists of two consecutive steps: (1) estimation of spatial extent of anomalies $l_{n,P}$ and (2) estimation of temporal extent d_n as shown in blocks III and IV in Figure 2, respectively. The spatial extent of delay $l_{n,P}$ is the length of the road segment that experiences anomalies, which can be estimated by finding a set of consecutive sensors where anomalies are detected in Section 4.3. As shown in Figure 4, the roadside sensors considered to be in the disruption area consist of *the start sensor, the intermediate sensors,* and *the end sensor*. The start sensor is the first downstream sensor where an anomaly is detected; that is, it is the location where delay is originated. The intermediate sensors are the sensors located consecutively upstream from the start sensor that also detects anomalies; that is, they signify the continuous upstream extension of anomalies. The end sensor is the first upstream sensor where an anomaly is not detected; that is, this is the location where the disruption ceases to exist and traffic upstream from this location is still under normal condition. Then, $l_{n,P}$ is estimated as the distance between the start and the end sensors.

The second step, temporal estimation of delay in block IV in Figure 2, uses $l_{n,P}$ as input to estimate delay vehicles are expected to experience on the road segment. Using the definition in [27], d_n, the estimated delay at time n, is calculated as the estimated additional travel time that exceeds the normal travel time as shown in (1), where $\tau_{n,P}$ is the estimated travel time and $\tau_{n,N}$ is the normal travel time on the same road segment. The estimated travel time $\tau_{n,P}$ is calculated by dividing $l_{n,P}$ with the estimated mean of speeds measured at the start, intermediate, and end sensors $v_{n,P}$.

The normal travel time $\tau_{n,N}$ is calculated by dividing $l_{n,P}$ with the normal speed $v_{n,N}$. We note that $v_{n,N}$ is calculated as the mean of measured speeds in the previous time window (size L) just before an anomaly is detected in the current time window (also size L). Therefore, $v_{n,N}$ reflects the expected speed drivers on the freeway segment would have experienced if an anomaly had not occurred.

$$d_n = \tau_{n,P} - \tau_{n,N}; \quad \tau_{n,P} = \frac{l_{n,P}}{v_{n,P}}; \ \tau_{n,N} = \frac{l_{n,P}}{v_{n,N}}. \qquad (1)$$

The estimated mean of speeds $v_{n,P}$ can be calculated as follows. Based on our previous study in [24] and further analysis of real-world data over a 6-month period, it is observed that the occurrences of delay are associated with five anomalous speed drop patterns, namely, linear, exponential, logarithmic, sigmoid, and unclassified. Furthermore, it is possible to find empirical equations that best fit each of these five speed drop patterns (e.g., an exponential equation that best fits the exponential speed drop patterns) [24]. A kernel function is used for the unclassified speed drop patterns.

For a given prediction horizon P and a total of M start, intermediate, and end sensors, these empirical equations can be used to temporally extrapolate speeds well into the future for each roadside sensor m, that is, $[v_{m,n+1}, v_{m,n+2}, \ldots, v_{m,n+P}]$. Then, $v_{n,P}$ can be obtained by first averaging temporally extrapolated speed samples $v_{m,n,P}$ and then averaging spatially across all M sensors as follows:

$$v_{n,P} = \frac{\sum_{m=1}^{M} v_{m,n,P}}{M}; \quad v_{m,n,P} = \frac{\sum_{i=1}^{P} v_{m,n+i}}{P}. \qquad (2)$$

In practice, delay estimation can be operated as follows. Once delay has been detected, speed patterns measured by each roadside sensor are measured and transmitted to the algorithm module in Figure 1. Using Dynamic Time Warping,

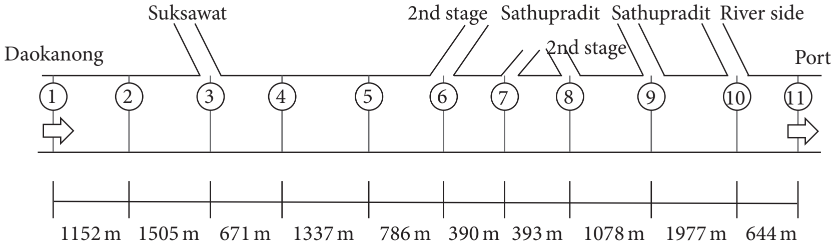


Figure 5: A sketch of expressway segment where real-world data was obtained.

these speed patterns are compared with the basis speed patterns associated with anomalies U_m^A, which have been collected and stored a priori, to find similarity. Once the most similar basis speed drop pattern has been found, its corresponding empirical equation is used to calculate the estimated mean speed $v_{n,P}$ as shown in (2), which is then used to calculate $\tau_{n,P}$ and eventually to estimate delay d_n in (1). Note that the calculation in (1) is for short-term delay [3]. Extending (1) to estimate vehicle's hour of delay or total delay experienced by all delayed vehicles on the road segment can be performed by adding traffic volume as a multiplication factor to (1).

5. Performance Evaluations

In this section, performance evaluations are conducted using real-world data. First, Section 5.1 presents two real-world data sets used to assess the proposed algorithm. Then, Section 5.2 describes an existing algorithm [17] which is used as benchmark. In Section 5.3, performance evaluation parameters are presented.

5.1. Descriptions of Real-World Data Sets

5.1.1. Real-World Data from Roadside Camera Sensors. The proposed algorithm is assessed using real-world data collected from a series of camera sensors installed on heavily used inbound section of Chalerm-Maha-Nakhon Expressway in Bangkok, Thailand. These sensors were part of a project jointly initiated by the Expressway Authority of Thailand (EXAT) and National Electronic and Computer Technology Center (NECTEC) aiming at real-time traffic monitoring. It consists of a series of eleven camera sensors installed consecutively along the 11 km expressway. Figure 5 shows the topology of the expressway and the locations of the camera stations, where the distance between two consecutive camera sensors ranges from 644 to 1977 meters. Each camera sensor is capable of detecting individual vehicles that passed the expressway segment and produced time series of traffic data including speed, occupancy, and flow in 1-minute intervals [34]. Each camera sensor is connected to a local EXAT operation office via wireless cellular network technologies (e.g., GPRS and 3G), where operators can view real-time traffic information and make decisions on traffic management.

A road segment within the full of view of a video camera is used as a designated area to calculate traffic data input. Speed is measured as a distance in a vehicle's traverse in the designated area divided by the time it takes. Flow is measured as the rate of vehicles passing a designated area at a given time interval. Occupancy is measured as a fraction of time that a designated area is occupied by vehicles.

Figure 6: An image of a traffic delay occurrence captured at the 4th camera sensor.

The data set used in this paper was collected from this sensor network over a one year period from January 1 to December 31, 2010. In this paper, we chose the 3rd, 4th, 5th, and 6th sensors as the experimental sites. An example of traffic congestion viewed by a camera sensor studied in this paper is shown in Figure 6. For evaluation purposes, anomalies and their associated delay that took place were independently and manually logged by a team of transportation experts. A total of 16 nonrecurring congestion cases were recorded which are used to test the anomaly detection capability of the proposed algorithm. Each record was associated with the time instances prior to the disruption occurrence and after it had ended. Besides these 16 cases, further investigation with another team of experts also revealed additional 44 delay cases. Therefore, a total of 60 delay cases are used to test the estimation operation of the proposed algorithm.

We note that the traffic data input of the proposed algorithm is typical data that most roadside sensors, for example, loop detectors, already produce. Therefore, we anticipate that if the sensors are to be replaced with other types of sensors, the performance of the system should not change significantly. This is worth further investigation for large scale deployment.

5.1.2. NGSIM Data. To show the potential of being deployed at a different location, the proposed algorithm is also assessed using NGSIM data [19]. The selected data set was collected on the southbound US Highway 101 in Los Angeles, California, USA, at 07:50 am–08:35 am. This data set is selected because it was collected during the buildup of congestion and consists of shockwaves which can potentially induce additional delay. It should be noted that the anomaly cases and the amount of delay caused by them were not labeled in this data set, so the full algorithm could not be deployed for this data set. However, the potential of the algorithm could be shown by shockwave detection. To assess the proposed algorithm, four virtual sensors are placed approximately 0.1 km apart to obtain traffic data to capture the occurrence of shockwaves and their spatial extents. Similar to conventional fixed traffic sensors (e.g., camera sensors and loop detectors), these virtual sensors calculate traffic data based on traffic flow and speed that pass the sensors' locations. An input time window size L of 5 minutes is used where time series of 30-second average speeds obtained from upstream and downstream virtual sensors are used as inputs. The occurrences of shockwaves are observed and recorded by a team of transportation researchers [25].

5.2. Benchmark Algorithm. To assess the capability of the proposed algorithm in respect to existing approaches, a benchmark algorithm is used [17]. This benchmark algorithm is selected because it has a similar objective to the proposed algorithm in detecting anomalies that cause delay. Also, this algorithm has already been implemented in a real-world anomaly detection system [17]. Furthermore, composed of a forecasting component and a detection logic component, it is designed to assess the difference between traffic data observed at consecutive roadside sensors.

However, the main difference between the benchmark algorithm and the proposed algorithm lies in the detection logic. The benchmark algorithm is designed to detect sudden significant changes of traffic inputs from the forecast component values by establishing threshold values (speed and occupancy) to filter out gradual changes in traffic inputs caused by recurring congestion, and an anomalous signal will only be raised when the specified threshold values have been exceeded. On the other hand, the proposed algorithm is designed to detect both sudden and gradual changes which enables it to detect different types of anomalies with fewer false alarms. This difference will become more apparent when both algorithms are assessed with real-world data in Section 6.

5.3. Performance Evaluation Parameters. To numerically evaluate the proposed algorithm, we use detection rate (DR), false positive rate (FPR), mean time to detection (MTTD), and mean absolute percentage error (MAPE) as shown in (3), (4), (5), and (6), respectively. DR, FPR, and MTTD are used to assess the anomaly detection operation of the proposed algorithm. MAPE is used to assess the accuracy of delay estimation of a total D recorded delay cases, where the error of the estimation $d_{n,j}$ of the jth recorded delay at time n is normalized by the actual delay $d^a_{n,j}$ to minimize the possibility of some estimated delay samples from dominating the others.

$$\mathrm{DR} = \frac{\text{Number of delays detected}}{\text{Total number of delays}}, \tag{3}$$

$$\mathrm{FPR} = \frac{\text{Number of detected delays that are false positives}}{\text{Total number of detected delays}}, \tag{4}$$

$$\mathrm{MTTD} = \frac{\text{Sum of time taken to detect individual delays}}{\text{Total number of detected delays}}, \tag{5}$$

$$\mathrm{MAPE} = \frac{1}{D}\sum_{j=1}^{D} \frac{\left|d^a_{n,j} - d_{n,j}\right|}{d^a_{n,j}}. \tag{6}$$

The performance parameters are applicable for data sets that contain labels for traffic anomalies and the amount of delay caused by the anomalies. Therefore, these parameters will be evaluated for the real-world data collected in Thailand, but, due to the lack of anomaly labels, could not be applied for the NGSIM data set.

6. Results and Discussions

In this section, the proposed algorithm is assessed using real-world data described in Section 5.1 and compared to the benchmark algorithm described in Section 5.2. Both algorithms are tested as if they operate online, where each detection operation is performed per input time window L.

6.1. Assessment Using NGSIM Data. The proposed algorithm is assessed on its capability to detect the the presence of shockwaves, which refers to a phenomenon where vehicles have to temporarily slow down below normal speed, and subsequently force the vehicles behind to slow down further. This phenomenon is known to create shockwave fronts or moving speed drop areas that move backward disrupting flow and annoying drivers behind [35]. The occurrence of shockwaves during congestion not only causes additional delay, but also increases chances of rear-end collisions. Identifying shockwaves during congestion also presents a challenge as it occurs temporarily and requires an algorithm that is fast and sensitive enough to detect them [25]. The occurrence of shockwaves represents unexpected events so they are considered as anomalies in this paper.

Figure 7 shows that the proposed algorithm can detect all four occurrences (Figure 7(a)) of shockwaves while benchmark algorithm can detect only two (Figure 7(b)). As shown in Figure 7(b), only abrupt changes in speed difference (second and third occurrences) can be detected by the benchmark algorithm. This algorithm is designed to detect consecutive differences between upstream and downstream traffic measurements so it operates well with consecutive changes in speed difference that are significant enough. On the other hand, benchmark algorithm misses the first and fourth occurrences of shockwaves in Figure 7(b) as they are associated with relatively more gradual changes of speed difference. Unlike benchmark algorithm, the proposed algorithm uses DTW to assess input patterns to detect delay occurrences so it can detect all four occurrences associated

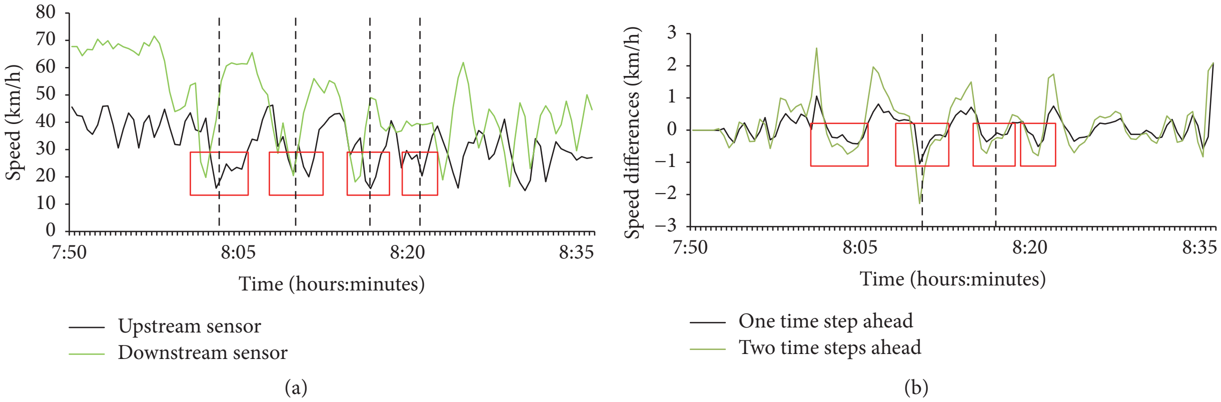


FIGURE 7: Detecting delay occurrences associated with shockwaves in NGSIM data using (a) the proposed algorithm and (b) benchmark algorithm [17]. Dashed vertical lines denote the time points where the algorithms detect delay occurrences. Red squares denote the duration where the occurrences of shockwaves are observed.

with both abrupt and gradual speed changes as shown in Figure 7(a).

Once the occurrences of shockwaves are detected, the proposed algorithm proceeds to estimate delay as described in Section 4.4. Using the spatial extent estimation described in Section 4.4, out of four sensors, three sensors are usually found to detect speed drops associated with shockwaves at a time. Therefore, the most downstream sensor and the most upstream sensor are selected as the start and end sensors, respectively, and $l_{n,P}$ is found to be 0.48 km.

Table 1 shows the values of $v_{n,P}$, $v_{n,N}$, $\tau_{n,P}$, and $\tau_{n,N}$ in (1) for each of the four shockwave occurrences. For comparison purposes, delay estimation is also applied for the second and third occurrences that are detected by benchmark algorithm (see Figure 7). Note that, since benchmark algorithm cannot detect the first and fourth occurrences, estimation operation is not activated during these times and normal speed and travel times are used instead. We further note that all four occurrences and their associated travel times and delays can be calculated consecutively online by the proposed algorithm.

As shown in Table 1, the four occurrences of shockwaves detected in Figure 7 are associated with travel times ($\tau_{n,P}$ in (1)) of 56.82 s, 55.54 s, 60.98 s, and 53.79 s, respectively. The proposed algorithm is able to detect all four occurrences and these estimated travel times would all be included in the delay calculation. On the other hand, if the benchmark algorithm was used instead, only the second and third occurrences and their associated travel times would be reported resulting in less accurate delay calculation. We note that, even though the difference between the delays obtained from the proposed algorithm and benchmark algorithm in this example may be small, significant improvement can be seen when higher degree of disruptions are assessed.

The results shown in Table 1 have many applications in advanced traveler information systems and advanced traffic management systems. A direct application would be to estimate travel time of the freeway segment and report it to motorists. The ability to detect occurrences of shockwaves can also be used in other applications such as calculating shockwave propagation time [25]. It can also be used to warn motorists to adjust intervehicle spacing to avoid rear-end collisions.

6.2. Assessment Using Real-World Data Collected in Thailand.

This section assesses the capability of the proposed algorithm and the benchmark algorithm [17] to detect the sixteen recorded traffic anomalies in the real-world data set described in Section 5.1. As inputs, both algorithms use speed, flow, and occupancy collected by roadside sensors. The assessment follows a k-fold cross-validation technique. The real-world data set consists of 16 anomaly cases $D_1, D_2, \ldots, D_{16}$ so evaluation is performed 16 times. At an iteration i, $i = 1, 2, \ldots, 16$, the anomaly case D_i is reserved as a test set, while the other anomaly cases are collectively used to calibrate the proposed algorithm and benchmark algorithm. That is, in the first iteration, $D_2, \ldots, D_{16}$ collectively serve as the training set, which is then tested with D_1. Then, in the second iteration, $D_1, D_3, \ldots, D_{16}$ are used for training, while D_2 is used for testing, and so on.

Table 2 shows that the proposed algorithm achieves higher detection rate and lower false alarm rate than the benchmark algorithm. Particularly, compared to benchmark algorithm, the proposed algorithm improves the detection rate and false positive rate by approximately 10% and 7%, respectively. Figures 8 and 9 show examples of anomalies detected by the proposed algorithm and found to cause traffic delay. Based on the records, these anomalies are associated with disabled vehicles parking over the shoulder area.

It is found that, for the benchmark algorithm, miss-detection usually occurs under low traffic flow. This algorithm always needs data from at least two sensors and the magnitude of traffic flow measured by the roadside sensors is not large enough to trigger the benchmark algorithm. While medium to heavy traffic flow triggers the benchmark algorithm, it also increases false alarms. This algorithm is designed to directly assess the difference between upstream and downstream sensors of the roadside sensors, which subsequently increases its sensitivity and triggers many false alarms.

Table 1: Estimation results on NGSIM data using the proposed algorithm and benchmark algorithm, $l_{n,P} = 0.48$ km. Note that if the shockwave is not detected, the estimated travel time $\tau_{n,P}$ would be equal to normal travel time $\tau_{n,N}$.

	1st	2nd	3rd	4th
Proposed algorithm				
Shockwaves detected	Yes	Yes	Yes	Yes
$v_{n,P}$ (see (1))	30.41 km/h	31.12 km/h	28.34 km/h	32.13 km/h
$\tau_{n,P}$ (see (1))	56.82 s	55.54 s	60.98 s	53.79 s
$d_n = \tau_{n,P} - \tau_{n,N}$	18.41 s	18.93 s	23.99 s	5.41 s
Benchmark algorithm [17]				
Shockwaves detected	No	Yes	Yes	No
$v_{n,P}$ (see (1))	44.99 km/h	31.12 km/h	28.34 km/h	35.72 km/h
$\tau_{n,P}$ (see (1))	38.41 s	55.54 s	60.98 s	48.38 s
$d_n = \tau_{n,P} - \tau_{n,N}$	0.00 s	18.93 s	23.99 s	0.00 s
If all 4 shockwaves are not detected				
$v_{n,N}$ (see (1))	44.99 km/h	47.21 km/h	46.71 km/h	35.72 km/h
$\tau_{n,N}$ (see (1))	38.41 s	36.60 s	37.00 s	48.38 s

Table 2: Results on anomaly detection using data set collected in Thailand.

Performance parameters	Proposed algorithm	Benchmark algorithm [17]
DR (%)	94	83
FPR (%)	5	12
MTTD (seconds)	340	140

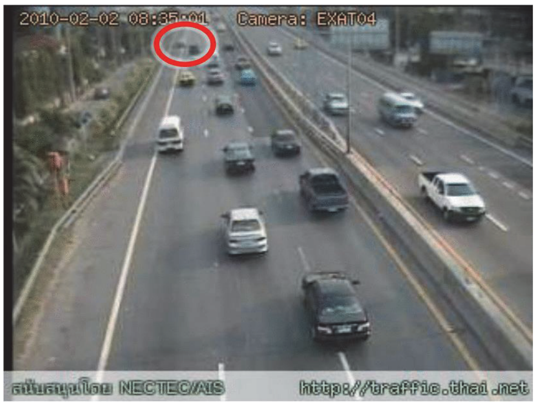


Figure 8: An anomaly detected at the 4th camera sensor.

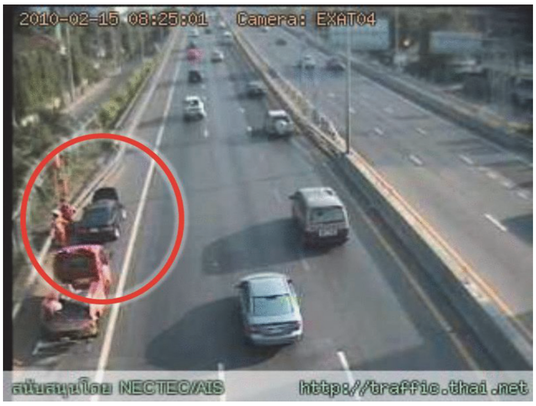


Figure 9: Another anomaly detected at the 4th camera sensor.

Unlike the benchmark algorithm, the proposed algorithm uses DTW to assess traffic data from each sensor individually. This enables the proposed algorithm to recognize patterns associated with anomalies more effectively, including under low traffic flow. The use of DTW also reduces a number of false alarms as it ensures that unrecognized patterns will not trigger detection at individual sensors. In respect to mean time to detection (MTTD), it takes longer for the proposed algorithm to detect anomalies compared to benchmark algorithm as it needs to wait till enough temporal samples of traffic data have been collected in each sensor before the detection operation can be triggered. However, the mean time to detection is still less than 6 minutes which should give enough time for traffic operators to response [8]. To reduce mean time to detection of the proposed algorithm, shorter time samples of traffic data can be used. Reducing MTTD while maintaining acceptable levels of DR and FPR is worth further investigation.

To deploy the proposed algorithm, computation time of the algorithm could be an added delay to the MTTD. Even though the computation time was not systematically measured for the performance evaluation, the computation time was in order of seconds. For the computer's specification that we use, which was a personal laptop with 2.5 GHz CPU and 4 GB of RAM, the computation time was approximately $[1, 3)$ seconds. As the computing time (a few seconds) was much smaller than the mean time to detection (up to 6 minutes), it is reasonable to consider the computing time negligible.

This section further discusses the results when the proposed algorithm was applied to estimate 60 delay cases described in Section 5.1, using different input time window sizes (L) and prediction horizons (P). For practical purposes, it is important to select suitable input time window size and prediction horizon to enable the proposed method to

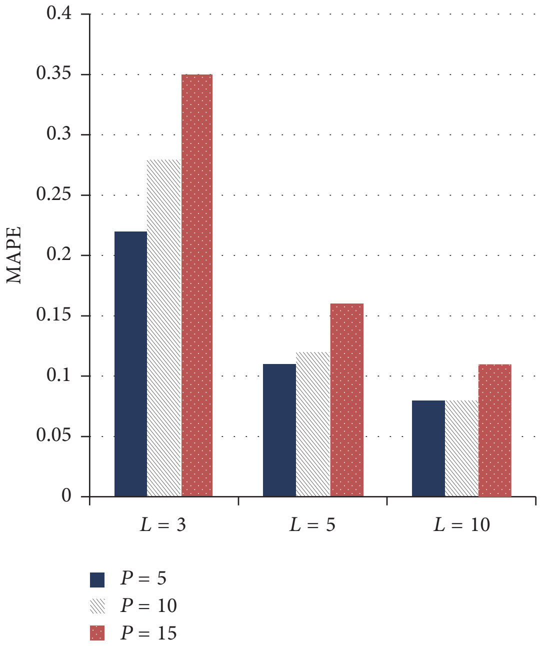


Figure 10: Mean absolute percentage error (MAPE) from estimating 60 delay cases using different input time window sizes L (minutes) and prediction horizons P (minutes).

perform effectively. Even though the 15-minute interval is often recommended [9] to avoid strong fluctuation, using lower input time window sizes can provide more fine-grained information for practitioners [25]. Therefore, in this paper, we use $L = 3$, 5, and 10 minutes for input time window size. For prediction horizons, we use $P = 5$, 10, and 15 minutes as this paper is focused on short-term estimation (less than 15 minutes). Also, changes in traffic condition on this experimental site are normally found to occur in less than 15 minutes.

The proposed algorithm is assessed on its capability to estimate delay using MAPE. Figure 10 shows MAPE from estimating 60 delay cases in the real-world data set with different input time window sizes (L) and prediction horizons (P). These results show that the estimation part of the proposed algorithm can achieve MAPE lower than 0.1 for this set of real-world data. This accuracy is achieved because the estimation part consists of both spatial and temporal estimations which enable the proposed algorithm to first estimate the area disrupted by anomalies and then estimate the delay itself. Furthermore, it can be seen that increasing input time window size (L) reduces MAPE as using more input samples enhances estimation accuracy. For the impact of prediction horizons (P), MAPE tends to increase with larger prediction horizons as it is more difficult to estimate delays farther ahead into the future. However, as shown in Figure 10, when P is increased, MAPE in fact increases at a much smaller rate when larger values of L are used. Therefore, it is recommended for practitioners that if the objective is to estimate delays far ahead into the future (using large prediction horizons), larger input time window size is needed.

Based on the findings in this section, it is important that practitioners select a suitable input time window size according to their practical objectives. If the objective is to provide prompt delay information, small input time window size should be used. On the other hand, if the objective is to provide accurately estimated delays for traffic management purposes, larger input time window sizes should be chosen. Furthermore, based on the proposed algorithm, a more complex and flexible system can also be further developed. For example, a stepwise approach can be implemented on top of the proposed algorithm where a small input time window size is used as a trigger to promptly discover the occurrence of traffic delay. Then, in the next step, the input time window size can be increased to collect more traffic data to achieve more accurate detection and estimation of traffic delay.

For anomalies with large scale effect beyond the 11 km experimental site, we anticipate that the proposed algorithm should be able to detect those anomalies if they occur downstream and cause changes in traffic patterns to propagate upstream into the 11 km segment. To accurately estimate delay, basis patterns associated with such large scale anomalies need to be incorporated to distinguish them. Then, an extrapolation technique needs to be incorporated to enable the proposed algorithm to estimate delay beyond the 11 km expressway segment where local traffic information cannot be directly observed [25]. This is worth further investigation.

7. Conclusions and Future Work

This paper proposes an algorithm to detect anomalies and estimate delay using traffic data from roadside sensors. The proposed algorithm uses traffic data from individual sensors to detect anomalies and then uses data from consecutive downstream and upstream sensors together to spatially and temporally estimate delay. Performance evaluations using real-world data show that the proposed algorithm achieves higher detection rate and lower false positive rate when used to detect anomalies compared to an existing benchmark algorithm. An investigation with NGSIM data also shows that the proposed algorithm is also able to detect the presence of shockwaves and, subsequently, estimate delay more efficiently than benchmark algorithm. Further results show that the estimation error increases at a much smaller rate when suitable input time window size and prediction horizon are selected.

There are interesting aspects that are worth further investigation. One interesting aspect is to reduce the processing time while maintaining low detection and estimation errors. A possible approach is to introduce a feature extraction method that can extract hidden patterns when smaller input time window sizes are used [8]. Alternative approaches include studying a set of hidden representative traffic flow features in advance by pretraining a deep-learning model [36] and populating traffic features into a genetic algorithm learning method to reduce dimensionality [37]. Another interesting aspect is to improve inference capability by incorporating a spatial inference method to estimate traffic data itself at the locations where they cannot be directly measured [25]. Also, it would be practically interesting to assess the proposed algorithm with different road types and geometries.

In addition to improvements of the algorithm, an interesting aspect to investigate is the applications of the algorithm. The solution provided in this paper is designed to be a streaming algorithm which can be run online; therefore, it can be naturally extended to use in various real-time applications. For example, the detected anomaly and estimated additional travel time would be useful for both traffic operators and travelers, if the information is delivered in a timely and convenient manner. Active communication between roadside sensors and vehicle-to-infrastructure (V2I) could be used to promptly transmit the information from our solution to nearby roadside message boards to display the information to the drivers on the roads or vehicles with V2I communication to adapt routing and navigation.

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgments

The authors would like to thank Dr. Supakorn Siddhichai and his IMG team at NECTEC for their assistance in providing real-world data for the analysis.

References

[1] N. Parrado and Y. Donoso, "Congestion based mechanism for route discovery in a V2I-V2V system applying smart devices and IoT," *Sensors*, vol. 15, no. 4, pp. 7768–7806, 2015.

[2] K. Nellore and G. P. Hancke, "A survey on urban traffic management system using wireless sensor networks," *Sensors*, vol. 16, no. 2, article 157, 2016.

[3] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: where we are and where we're going," *Transportation Research—Part C: Emerging Technologies*, vol. 43, pp. 3–19, 2014.

[4] R. Li and G. Rose, "Incorporating uncertainty into short-term travel time predictions," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 6, pp. 1006–1018, 2011.

[5] Y. Chung, "A methodological approach for estimating temporal and spatial extent of delays caused by freeway accidents," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1454–1461, 2012.

[6] J. Weng and Q. Meng, "Estimating capacity and traffic delay in work zones: an overview," *Transportation Research—Part C: Emerging Technologies*, vol. 35, pp. 34–45, 2013.

[7] X. Zhang, E. Onieva, A. Perallos, E. Osaba, and V. C. S. Lee, "Hierarchical fuzzy rule-based system optimized with genetic algorithms for short term traffic congestion prediction," *Transportation Research—Part C: Emerging Technologies*, vol. 43, pp. 127–142, 2014.

[8] S. Thajchayapong, E. S. Garcia-Trevino, and J. A. Barria, "Distributed classification of traffic anomalies using microscopic traffic variables," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 448–458, 2013.

[9] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 871–882, 2013.

[10] F. Zheng and H. Van Zuylen, "Urban link travel time estimation based on sparse probe vehicle data," *Transportation Research—Part C: Emerging Technologies*, vol. 31, pp. 145–157, 2013.

[11] F. Dion and H. Rakha, "Estimating dynamic roadway travel times using automatic vehicle identification data for low sampling rates," *Transportation Research Part B: Methodological*, vol. 40, no. 9, pp. 745–766, 2006.

[12] D. Paterson and G. Rose, "Dynamic travel time estimation on instrumented freeways," in *Proceedings of the 6th World Congress on Intelligent Transportation Systems*, Toronto, Canada, 1999.

[13] J. Kwon, B. Coifman, and P. Bickel, "Day-to-day travel-time trends and travel-time prediction from loop-detector data," *Transportation Research Record*, no. 1717, pp. 120–129, 2000.

[14] J. Rice and E. Van Zwet, "A simple and effective method for predicting travel times on freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 3, pp. 200–207, 2004.

[15] C. D. R. Lindveld, R. Thijs, P. H. L. Bovy, and N. J. van der Zijpp, "Evaluation of online travel time estimators and predictors," *Transportation Research Record*, vol. 1719, pp. 45–53, 2000.

[16] H. Yang, S. Chen, Y. Wang, B. Wu, and Z. Wang, "Comparison of delay estimation models for signalised intersections using field observations in Shanghai," *IET Intelligent Transport Systems*, vol. 10, no. 3, pp. 165–174, 2016.

[17] C. L. Mak and H. S. L. Fan, "Development of dual-station automated expressway incident detection algorithms," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 3, pp. 480–490, 2007.

[18] K. Hi-Ri-O-Tappa, C. Likitkhajorn, A. Poolsawat, and S. Thajchayapong, "Traffic incident detection system using series of point detectors," in *Proceedings of the 15th International IEEE Conference on Intelligent Transportation Systems (ITSC '12)*, pp. 182–187, Anchorage, Alaska, USA, September 2012.

[19] V. Alexiadis, J. Colyar, J. Halkias, R. Hranac, and G. McHale, "The next generation simulation program," *ITE Journal*, vol. 74, no. 8, pp. 22–26, 2004.

[20] F. G. Habtemichael, M. Cetin, and K. A. Anuar, "Incident-induced delays on freeways: quantification method by grouping similar traffic patterns," *Transportation Research Record*, vol. 2484, pp. 60–69, 2015.

[21] J. Li, C.-J. Lan, and X. Gu, "Estimation of incident delay and its uncertainty on freeway networks," *Transportation Research Record*, vol. 1959, pp. 37–45, 2006.

[22] Y. Chung, "Assessment of non-recurrent congestion caused by precipitation using archived weather and traffic flow data," *Transport Policy*, vol. 19, no. 1, pp. 167–173, 2012.

[23] J. A. Barria and S. Thajchayapong, "Detection and classification of traffic anomalies using microscopic traffic variables," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 695–704, 2011.

[24] K. HiriOtappa and S. Thajchayapong, "Generalizability and transferability of incident detection algorithm using dynamic time warping," in *Proceedings of the in 19th World Congress on Intelligent Transportation Systems*, Vienna , Austria, 2012.

[25] S. Thajchayapong and J. A. Barria, "Spatial inference of traffic transition using micro–macro traffic variables," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 854–864, 2015.

[26] C. A. Ratanamahatana and E. Keogh, "Everything you know about dynamic time warping is wrong," in *Proceedings of the 3rd Workshop on Mining Temporal and Sequential Data in Conjunction with the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '04)*, Citeseer, Seattle, Wash, USA, August 2004.

[27] Transportation Research Thesauras, Transportation Research Board National Academy, http://trt.trb.org/trt.asp?NN=Bthd.

[28] N. Chiabaut, C. Buisson, and L. Leclercq, "Fundamental diagram estimation through passing rate measurements in congestion," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 355–359, 2009.

[29] J. J. Fernández-Lozano, M. Martín-Guzmán, J. Martín-Ávila, and A. García-Cerezo, "A wireless sensor network for urban traffic characterization and trend monitoring," *Sensors*, vol. 15, no. 10, pp. 26143–26169, 2015.

[30] J.-S. Oh, C. Oh, S. G. Ritchie, and M. Chang, "Real-time estimation of accident likelihood for safety enhancement," *Journal of Transportation Engineering*, vol. 131, no. 5, pp. 358–363, 2005.

[31] A. P. Chassiakos and Y. J. Stephanedes, "Smoothing algorithms for incident detection," *Transportation Research Record*, vol. 1394, pp. 8–16, 1993.

[32] V. Alarcon-Aquino and J. A. Barria, "Anomaly detection in communication networks using wavelets," *IEE Proceedings—Communications*, vol. 148, no. 6, pp. 355–362, 2001.

[33] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proceedings of the Workshop on Knowledge Discovery in Databases (KDD '94)*, vol. 10, pp. 359–370, Seattle, Wash, USA, August 1994.

[34] K. Kiratiratanapruk and S. Siddhichai, "Vehicle detection and tracking for traffic monitoring system," in *Proceedings of the IEEE Region 10 Conference, (TENCON '06)*, pp. 1–4, IEEE, Hong Kong, November 2006.

[35] A. Hegyi, B. De Schutter, and J. Hellendoorn, "Optimal coordination of variable speed limits to suppress shock waves," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 102–112, 2005.

[36] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[37] P. Lopez-Garcia, E. Onieva, E. Osaba, A. D. Masegosa, and A. Perallos, "A hybrid method for short-term traffic congestion forecasting using genetic algorithms and cross entropy," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 557–569, 2016.

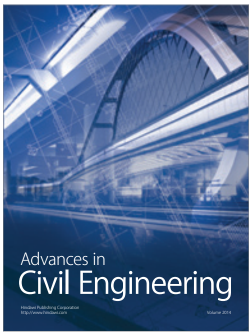
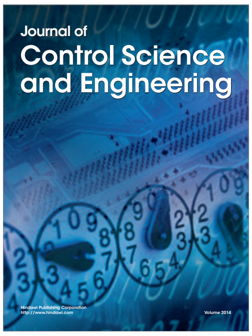
Journal of
Engineering

The Scientific
World Journal

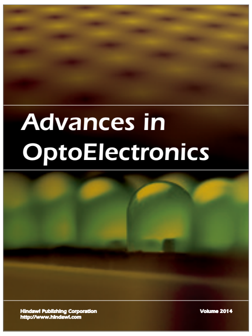
International Journal of
Rotating
Machinery

Journal of
Sensors

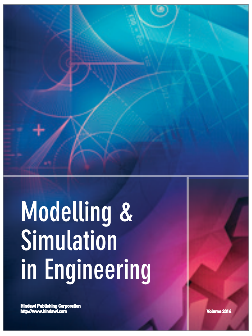
International Journal of
Distributed
Sensor Networks

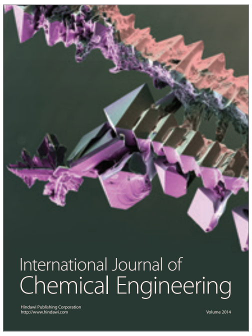
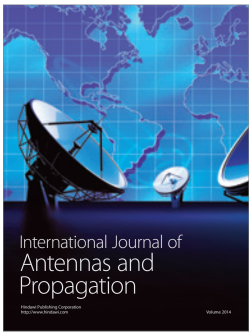

Advances in
Civil Engineering

Journal of
Control Science
and Engineering

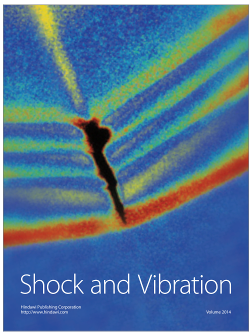
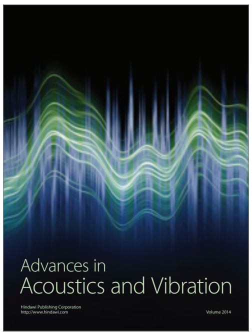
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

Advances in
OptoElectronics

VLSI Design

International Journal of
Navigation and
Observation

Modelling &
Simulation
in Engineering

International Journal of
Aerospace
Engineering

International Journal of
Chemical Engineering

International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components

Shock and Vibration

Advances in
Acoustics and Vibration