Data-driven parallelizable traffic incident detection using spatio-temporally denoised robust thresholds

Pranamesh Chakraborty[a,*], Chinmay Hegde[b], Anuj Sharma[a]

[a]*Civil, Construction, and Environmental Engineering Department, Iowa State University, Ames, Iowa, USA 50011*
[b]*Electrical and Computer Engineering Department, Iowa State University, Ames, Iowa, USA 50011*

Abstract

Automatic incident detection (AID) is crucial for reducing non-recurrent congestion caused by traffic incidents. In this paper we propose a data-driven AID framework that can leverage large-scale historical traffic data along with the inherent topology of the traffic networks to obtain robust traffic patterns. Such traffic patterns can be compared with the real-time traffic data to detect traffic incidents in the road network. Our AID framework consists of two basic steps for traffic pattern estimation. First, we estimate robust univariate speed threshold using historical traffic information from individual sensors. This step can be parallelized using MapReduce framework thereby making it feasible to implement the framework over large networks. Our study shows that such robust thresholds can improve incident detection performance significantly compared to traditional threshold determination. Second, we leverage the knowledge of the topology of the road network to construct threshold heatmaps and perform image denoising to obtain spatio-temporally denoised thresholds. We used two image denoising techniques, bilateral filtering and total variation for this purpose. Our study shows that overall AID performance can be improved significantly using bilateral filter denoising compared to the noisy thresholds or thresholds obtained using total variation denoising.

Keywords: freeway incident detection, threshold denoising, total variation, bilateral filter, mapreduce

1. Introduction

Traffic congestion in freeways poses a major threat to the economic prosperity of the nation (Owens et al., 2010). Freeway congestion is usually classified into two categories: recurrent congestion and non-recurrent congestion (Ozbay and Kachroo, 1999; Dowling et al., 2004; Anbaroglu et al., 2014). Recurrent congestion typically exhibits a daily pattern, observed in morning or evening peaks. On the other hand, non-recurrent congestion are mainly caused by unexpected events like traffic incidents or stalled vehicles (Anbaroglu et al., 2014). Such events are a major source of travel time variability (Noland and Polak, 2002) and often cause frustration to the commuters (FHWA). Hence, automatic incident detection (AID) has been identified to be a crucial technology for reduction of non-recurrent traffic congestion (Sussman, 2005). Early incident detection has shown to save 143.3 million man-hours and $3.06 million in 2007 (Schrank and Lomax, 2007). Consequently, significant research has been done on development of accurate AID algorithms.

Various data-driven algorithms and statistical models have been used to develop AID algorithms. A popular approach for detecting traffic incidents is to learn the traffic patterns using accumulated traffic data observed in the past, and detect incidents when traffic data observed in real-time behaves significantly different from the learned patterns (Dudek et al., 1974; Kamran and Haas, 2007; Zhang et al., 2016). With the recent advancement in traffic data collection and data storage technologies, fixed sensors installed on roads or probe vehicles can provide useful information on the real-time traffic state over vast networks. These data sources when compared and matched with corresponding historical datasets can serve as useful indicators of traffic incidents.

*Corresponding author (pranames@iastate.edu)

However, two major challenges arise in such AID algorithm development. First, it is very difficult to obtain a consistent statistical model of the dynamics of traffic patterns. Recurring congestion events can pair with the non-recurring events at the same time and location, thereby making it difficult to separate the true positives (incidents) from false positives (caused by recurring congestion). Second, with the recent advancements in data storage technologies, the scale of traffic data stored from a traffic network makes it difficult to process it in real-time and detect traffic incidents. Considerable amount of computation is involved in parsing the data which makes it difficult for conventional statistical methods to handle such massive data sources and enable a real-time AID algorithm.

The primary objective of this paper is to propose and implement a *massively parallelizable* data-driven framework for AID in freeways. Our framework leverages the traffic information obtained from each sensor, together with the inherent topology of the traffic network to obtain robust traffic pattern estimates. Such traffic patterns can be used to detect traffic incidents by comparing the real-time traffic data with the corresponding pattern and detecting the anomalies. Further, such a framework can be easily extended over large highway networks due to the its inherent parallelizable framework.

The primary conceptual building block of our approach is to sub-divide the road network into multiple smaller segments. Each segment produces a time-series of the traffic state (average speed). These time-series are extensively large-scale, with thousands of data points being recorded daily per segment. In order to enable real-time computation, we first perform dimensionality reduction with novel *robust summary statistics* of each time-series across non-overlapping time windows. This summary statistics computation can be massively parallelized using MapReduce (Dean and Ghemawat, 2008), thereby making it feasible to apply the framework over large networks. We develop on our previous work Chakraborty et al. (2017b) to show how these robust summary statistics are less susceptible to optimum threshold determination compared to traditional standard normal deviates based statistics. However, this summary statistics computation do not take into account the spatio-temporal correlations of the time windows, and are therefore noisy. To resolve this, we leverage the knowledge of the topology of the road-network and construct a "heatmap" of the summary statistics. We then perform multivariate denoising of the "heatmap" assuming that the summary statistics of the topologically and temporally adjacent regions are likely to be similar. Recently, Chakraborty et al. (2017a) applied trend filtering based methods for denoising the threshold heatmaps to be used for incident detection. However, the study didn't covered what benefits can be achieved in AID performance using the denoised heatmaps. In this study, we used bilateral filtering for threshold denoising and compared the performance with total variation, used by Chakraborty et al. (2017a), to show the improvements obtained in incident detection performance with such spatio-temporally denoised robust summary statistics.

The paper is organized as follows. Section 2 provides a brief description on the relevant literature on traffic incident detection in freeways. Section 3 describes the methodology adopted in this study followed by the data description in Section 4. Section 5 provides the details of the results obtained using the proposed methodology. Finally, Section 6 provides a summary of our results and points to directions of future study.

2. Literature Review

Freeway traffic incidents are usually classified as anomalies or outliers in the traffic stream. AID algorithms aim to detect such anomalies using real-time traffic data along with historical data (whenever available). AID algorithms can be primarily classified into two categories based on the methodology adopted:

1. Real-time traffic data is compared with the traffic data observed in the immediate past (i.e., over the previous T intervals) to detect abnormalities which can be classified as incidents.
2. Historical traffic data is used to generate "normal" traffic patterns, and significant deviation from the normal patterns are classified as incidents.

Several existing algorithms compare real-time traffic conditions with immediate past conditions to detect traffic incidents. Parkany and Bernstein (1995) detected traffic incidents based on the principle that when traffic conditions switches from incident-free to incident conditions, frequent lane switch maneuvers and temporal and spatial discrepancies in headway and travel conditions can be observed. Hellinga and Knapp (2000) proposed the Waterloo algorithm where it is assumed that travel-time is log-normally distributed and normal travel pattern is estimated from the traffic data of past T intervals. Li and McDonald (2005) proposed the bivariate analysis model (BEAM) where

travel time difference between adjacent travel time intervals are used for detecting incidents. On the other hand, Zhu et al. (2009) used both spatially and temporally adjacent time intervals as features for incident detection purposes. Li et al. (2013) used weighted average and standard deviation of past T intervals of traffic parameter values to detect traffic incidents from the data stream. False alarms due to fluctuations in traffic variables were handled by replacing the mean and standard deviation of current time intervals with those of the previous time intervals when coefficient of variation of traffic variable was found to be less than a predetermined threshold. Recently, Asakura et al. (2017) used shockwave theory to probe-vehicle trajectories to detect traffic incidents. Although all the above algorithms have been used extensively for incident detection, they do not utilize the wealth of information available from past historical data. Rather, they rely only on the real-time and the immediate past traffic data to detect the abnormalities in the data stream.

Using only real-time and immediate past traffic data for incident detection helps in avoiding the issues in storing and processing large scale traffic data. However, with the recent advancements in data storage and data processing technologies (Zhang et al., 2017), it makes more sense to utilize the information from historical traffic data to develop efficient AID algorithms. Along. For example, Sethi et al. (1995) used linear discriminant analysis to learn the linear relationship of predictor variables from historical traffic data that can differentiate between incident and non-incident conditions. Balke et al. (1996) used standard normal deviates (SND) to generate the confidence intervals for normal traffic conditions. Significant deviations from normal conditions can be labeled as traffic incidents. Historic average travel times and their normal deviates were determined by time of day and day of week for each road segment to denote incident-free conditions. Since mean and standard normal deviates are known to be prone to outliers, Balke et al. (1996) used an incident dataset for removing the outliers from historic data before summary statistics computation. So, this method also requires traffic incident dataset with accurate start and end time of the incidents (which are often hard to get) for incident detection. Snelder et al. (2013) used median instead of mean in order to achieve *robust* summary statistics computation and used it as a reference case to find out delays caused by incidents.

SND and such similar methods do not take into consideration the spatio-temporal relationship of the road network to generate the normal traffic conditions. However, Chung (2011) applied SND over spatio-temporally connected cells to determine incident-induced delay. Chung and Recker (2012) further extended the method using an optimization method to determine optimal hyper-parameter c in SND method ($\mu - c \times \sigma$, where μ denotes the mean and σ denotes the standard deviation) to estimate the spatio-temporal extent of incidents. These studies however do not consider the spatio-temporal correlation of the thresholds. They rather use the topological and temporal information to determine impact region of traffic incidents based on the thresholds generated from univariate time-series. Similarly, Anbaroglu et al. (2014) used spatio-temporal clustering for detecting non-recurrent traffic congestion when link journey time of spatio-temporally connected cells exceed pre-determined threshold computed from historical data. Chen et al. (2016) also used spatio-temporal clustering to identify recurrent and non-recurrent congestion, their spatio-temporal extent and quantify the delay caused by such incidents. Similarly, Zhang et al. (2016) used dictionary-based compression theory for identifying the spatio-temporal traffic patterns at detector, intersection, and sub-region level which can be used for anomaly identification.

Thus, significant research has been conducted to develop accurate AID algorithms using the rich information obtained from historical traffic data. Spatio-temporal analysis has also been performed to find out the extent of traffic incidents and their impact regions. This study complements the existing research by providing extending the AID framework in two different ways. The first part involves computation of robust summary statistics using parallel computation methods enabling application of the AID algorithm over large-scale traffic network. The second part involves generation of accurate summary statistics by multivariate denoising which utilizes the spatio-temporal correlation of the parameters obtained from the first part. Overall, we show that our proposed AID framework can achieve improved overall detection performance.

3. Methodology

Lane-blocking traffic incidents or incidents impacting the traffic stream often result in significant drop in vehicle speeds and/or increase in the traffic densities. Hence AID algorithms often model traffic incidents as outliers or anomalies in the traffic data stream. The basic objective of the AID algorithm is to detect these anomalies by comparing the real-time traffic data with the immediate past data or with the historical past data. In this study, we extend the popular SND algorithm (Dudek et al., 1974; Balke et al., 1996) of incident detection with uses the historical traffic

data to learn traffic pattern. Then the traffic pattern is compared with real-time traffic data to detect the anomalies in the streaming real-time data. Our incident detection framework consists of two steps:

1. *Univariate speed threshold determination:* This involves determination of robust summary statistics (thresholds) of each univariate time series resulting from each road-segment.
2. *Multivariate spatio-temporal threshold denoising:* The thresholds determined in the previous step are now denoised using the spatio-temporal correlations of the adjacent thresholds.

Next, we set up our model, followed by a detailed description of each of the above two steps.

3.1. Setup

We first describe the mathematical abstraction of our data corpus. Let the weighted graph of the traffic network is denoted by $G = (S, E, W)$. Here, $S = \{s_i\}_{i=1}^{n}$ denotes the nodes of the graph, $E = \{e_i\}_{i=1}^{m}$ denotes the (undirected) edges, and $W = \{w_i\}_{i=1}^{m}$ denotes the weights of the graph. In our framework, the nodes of the graphs correspond to the consecutive road-segments into which the freeway under consideration is partitioned, and an average vehicle speed (x) is reported each minute for each segment. The nodes corresponding to consecutive segments along the freeway are connected via appropriately weighted edges. In this study, we assume that the weights of each node are equal to 1 (i.e., unweighted edge segments) but conceptually these can be used to encode other spatial information. The topology of the road network (i.e., the order of the road segments) describes the connectivity of our graph system. We measure a (noisy) time series for each node s on a given day (say d) of length N:

$$x_s^d = \left(x_s^{t_1,d}, x_s^{t_2,d}, ..., x_s^{t_N,d}\right) \tag{1}$$

Here, t_i denotes the i^{th} time instant, and d denotes the day of the week. The observed time series are synchronized across different nodes, i.e., the time stamps $t_1, t_2, ..., t_N$ are same for all days across all segments.

Overall, we model the time series as a *third-order tensor* $x \in \mathbb{R}_+^{n \times N \times D}$. Here, D denotes different days, n refers to the total number of nodes in the graph, and N denotes the length of the time series. For example, if average speed of each segment is reported in 1-minute intervals (as in this study), then the length of the time series N will be equal to $24 \times 60 = 1440$. Our objective here is to identify anomalous local patterns in this tensor.

However, a major challenge that we face is the scale of the traffic data. The number of road segments and the sampling rate of the segments are very high, thereby producing millions of traffic records daily. For example, the entire road network of Iowa, divided into approximately 54,000 segments, produce 4 gigabytes of daily traffic speed data, which aggregates to approximately 1.5 terabytes of annual traffic data.

To alleviate this issue, we first pre-process along the second dimension of the tensor. We perform robust summary statistics computation of each univariate time series across non-overlapping windows to determine (scalar) thresholds of each window in the time series, which will serve as key parameters in our AID framework. The details of this step is discussed next.

3.2. Univariate Threshold Computation

Our basic methodology for univariate threshold computation is based on the popular SND algorithm (Dudek et al., 1974; Balke et al., 1996) for incident detection. The algorithm involves modeling the univariate statistics of each non-overlapping windows of the time series data as a Laplace distribution with location parameter μ and scale parameter ζ. More specifically, each univariate time series is divided into 15-minute non-overlapping windows (p), similar to study by Dudek et al. (1974), and the threshold speed (τ) for each window is determined from the location and scale parameters. The threshold ($\tau_s^{p,d}$) speed is defined as:

$$\tau_s^{p,d} = \mu_s^{p,d} - c \times \zeta_s^{p,d} \tag{2}$$

where, the optimum constant c is to be determined from the validation set. This auxiliary reduced tensor, speed threshold ($\tau_s^{p,d}$), can be compared with real-time speed data $X_s^{t_i,d}$ to detect anomalies or traffic incidents.

The SND algorithm uses the mean speed value ($\bar{x}$) as the location parameter (μ) and the corresponding standard deviation (σ) as the scale parameter (ζ) to model the normal traffic pattern. Normal traffic condition varies depending on the time of day and day of the week. Hence the location and scale parameters are determined for each day of the

week, and 15-minute periods of the day for each segment. Threshold speed values over 15-minute intervals of each day of the week are determined using the previous 8 weeks of traffic data for the same segment for the given day of the week and period of the day, similar to the study of Balke et al. (1996). The mean speed values ($\bar{x}_s^{p,d}$), standard deviation $\sigma_s^{p,d}$, and threshold speed $\tau_{s,snd}^{p,d}$ for SND algorithm can be determined as follows:

$$\bar{x}_s^{p,d} = \frac{\sum_{\forall k} x_{k,s}^{p,d}}{\sum_{\forall k} k} \tag{3}$$

$$\sigma_s^{p,d} = \frac{1}{\sum_{\forall k} k} \sum_{\forall k \in (d,p,s)} \left(x_{k,s}^{p,d} - \bar{x}_s^{p,d} \right)^2 \tag{4}$$

$$\tau_{s,snd}^{p,d} = \bar{x}_s^{p,d} - c_{snd} \times \sigma_s^{p,d} \tag{5}$$

Although the SND algorithm is easy to calibrate and inherits good transferability (Li et al., 2013), a major drawback of the algorithm is that it is prone to outliers or anomalies (Balke et al., 1996). Both location and scale parameters (mean and standard deviation) are known to be highly susceptible to outliers (Pearson, 2005). This causes a severe challenge for traffic datasets that contain anomalies (in the form of incidents), and can lead to misrepresentation of the underlying normal traffic patterns due to these anomalies. One possible way to tackle this issue is to manually remove all time intervals that are known incidents (e.g. by referring to incident reports), before calculating the aforementioned threshold parameters. This requires complete knowledge of the traffic incident data during both training, testing, and implementation; however, it is often difficult to get accurate reports of incidents (Ren et al., 2012). In particular, it is hard to obtain accurate estimates for start time, duration, and impacted regions due to traffic incidents (Yue et al., 2016). To alleviate this issue, we propose to use *alternate* robust summary statistics (learned from the time series tensor itself) for the location and scale parameters so that the affect of the presence of anomalies can be minimized.

Outlier detection is a important task in statistical analysis and significant research has been performed for development of robust models to detect outliers from noisy data streams (Pearson, 2005; Aggarwal, 2007). Interested readers can refer to Gupta et al. (2014) for a detailed review of current outlier detection methods. In this study, we propose to use two such modifications of the SND summary statistics to calculate robust location and scale parameters. Specifically, we replace the standard deviation values by maximum absolute deviation (MAD) (Hampel, 1974) and inter-quartile deviation (IQD) for the scale parameter (ζ). For both these methods, we use median speed values instead of mean values as location parameter (μ). Studies have shown that median, IQD, and MAD provide more robust summary statistics compared to mean and standard deviation for data containing outliers (Pearson, 2005; Leys et al., 2013). Thus, the two modified univariate threshold computation techniques can now be expressed as:

1. MAD algorithm: μ = Median (M), ζ = Maximum Absolute Deviation (MAD),

$$\tau_{s,mad}^{p,d} = M_s^{p,d} - c_{mad} \times MAD_s^{p,d} \tag{6}$$

2. IQD algorithm: μ = Median (M), ζ = Inter-Quartile Distance (IQD),

$$\tau_{s,iqd}^{p,d} = M_s^{p,d} - c_{iqd} \times IQD_s^{p,d} \tag{7}$$

The median (M), maximum absolute deviation (MAD), and inter-quartile deviation (IQD) can be calculated as follows:

$$M_s^{p,d} = median_{\forall k} \left(x_{k,s}^{p,d} \right) \tag{8}$$

$$MAD_s^{p,d} = median_{\forall k} \left| x_{k,s}^{p,d} - M_s^{p,d} \right| \tag{9}$$

$$IQD_s^{p,d} = \left\{ x_{k,s}^{p,d} : \Pr\left(X \leq x; \forall k\right) = 0.75 \right\} - \left\{ x_{k,s}^{p,d} : \Pr\left(X \leq x; \forall k\right) = 0.25 \right\} \tag{10}$$

Although MAD, and IQD are known to be robust to outliers, one of the major drawbacks of these statistics are they are susceptible to "swamping" problems where non-outliers are classified as outliers resulting in significant number of false alarms. This implies that if more than 50% of the data values (x_k) are very similar (i.e., close to swamping breakdown point), then IQD and MAD are both equal to zero. So, any value different from median will be reported as

an outlier. This has important consequences for incident detection because traffic data is usually concentrated along a particular value. For example, a freeway segment with speed limit of 70 mph will mostly report speed values around 70 mph. So, when more than 50% of speed values are equal to 70 mph, the ζ parameter (i.e., MAD or IQD) will be equal to zero and hence a speed value of even 69 mph will be reported as an outlier. This will result in significant increase in the number of false alarms.

However, in traffic incident detection problems, we can take advantage of the fact that the capacity-reducing traffic incidents will significantly impact the traffic conditions and result in congested traffic conditions. To recall, AID algorithms relying on macroscopic traffic data solely (instead of camera, probe vehicle trajectories or similar data sources) can only detect incidents which impact traffic flow and result in significant reduction in the capacity. So, the incident alarm can be triggered only when congested conditions exist and the observed speed is lower than the expected threshold (given by Equation 2). Federal Highway Administration guidelines state that congested conditions occur in freeways when average speed in a road-segment is less than 45 mph (Systematics, 2005). So, the modified threshold speed value can now be written as:

$$\tau_s^{p,d} = Min\left[45, \mu_s^{p,d} - c \times \zeta_s^{p,d}\right] \tag{11}$$

One of the major advantage of these summary statistics (mean, median, MAD, and IQD) is that their computation can be easily parallelized over multiple systems thereby enabling the AID framework to handle massively large datasets over wide traffic network. In this study, we used Apache Pig (2018), a Hadoop MapReduce framework for computation of the summary statistics values from raw traffic speed data. The optimal constant parameters, c_{snd}, c_{mad}, and c_{iqd}) in Equations 5, 6, and 7 respectively are determined using the incident validation data set. Also, similar to past studies (Ren et al., 2012; Li et al., 2013), we perform *persistence test* before triggering incident alarm. This means, incident alarm is triggered when observed speed values are lower than the threshold speed value 11 for three consecutive intervals. This is done to reduce false alarms due to spurious noisy low speed values.

3.3. Multivariate Spatio-Temporal Threshold Denoising

Univariate threshold computation, given by Equation 11, does not take into consideration the topology of the traffic network or temporal correlations between the time windows. Hence, the thresholds computed can be highly noisy and variable across contiguous roadway segments. To compensate for this, we propose to improve the quality of the estimated thresholds using the spatio-temporal information. We leverage both the underlying topology of the road network and the temporal coherence of the time windows, and perform denoising to obtain improved estimates of the speed threshold values depicting the normal traffic pattern.

First, we obtain a "heatmap" of the threshold speeds calculated using Equation 11. Figure 1 shows a sample speed threshold heatmap. The road segments are arranged according to their corresponding mileage, along with temporally consecutive time windows to form the heatmap. We argue that the spatially and temporally coherent time windows are likely to exhibit similar thresholds. So, we formulate our objective as to obtain a coherent, smoothed threshold heatmap by performing denoising of the raw heatmap. In this study, we used two specific procedures of image denoising techniques — bilateral filter and total variation — to obtain the denoised threshold map which can be used for improved incident detection performance. We chose these denoising techniques since they are known to preserve the edges during denoising. This is important given the fact the sharp edges in threshold heatmaps often indicate the regions of recurrent congestion (as shown in Figure 1) and preserving the edges will help to differentiate the recurrent congestion from non-recurrent congestion events such as traffic incidents. We discuss each of the two threshold denoising procedures next.

3.3.1. Bilateral Filter based Threshold Denoising

Bilateral filtering (Tomasi and Manduchi, 1998) is a popular image denoising technique which preserves edges while smoothing. In the most basic formulation, each pixel in an image is replaced by the average of its neighbors, keeping into account the geometric closeness (spatial and temporal correlations) along with the photometric similarity (speed threshold values). As stated in Section 3.1, the multivariate time series can be represented as a third-order tensor $x \in \mathbb{R}_+^{n \times N \times D}$, where D denotes different days of the week (producing different heatmaps), n refers to the total number of nodes in the graph, and N denotes the length of the time series. So, each threshold heatmap can be

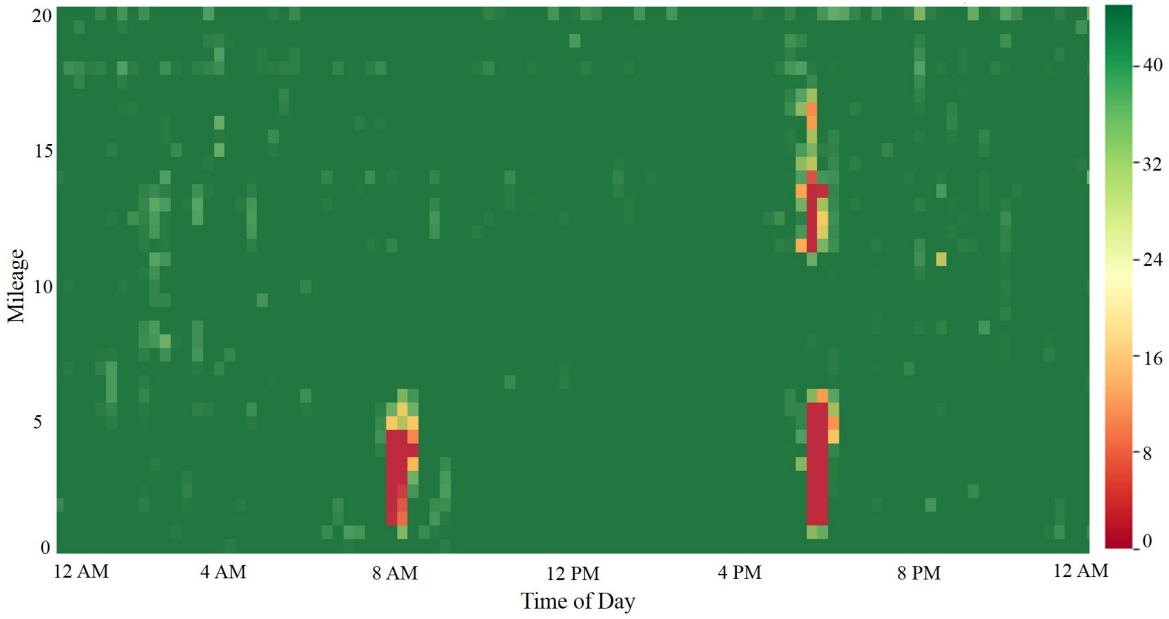


Figure 1: Sample speed threshold heatmap (mph)

represented as an image (a second-order tensor, $I : \tau \in \mathbb{R}_{+}^{n \times N'}$). Since we divide the time of the day into 15-minute time windows, so $N' = (60/15) \times 24 = 96$.

The Bilateral Filter, denoted by $BF[.]$, can be defined as a function that takes in an image I as input, and returns an image $BF[I]$ whose p^{th} pixel intensity is given by:

$$BF[I]_p = \frac{\sum\limits_{q \in S} G_{\sigma_s}(\|p-q\|) G_{\sigma_r}\left(I_p - I_q\right) I_q}{\sum\limits_{q \in S} G_{\sigma_s}(\|p-q\|) G_{\sigma_r}\left(I_p - I_q\right)} \tag{12}$$

where S and R denotes the space domain (set of possible pixel locations in an image) and range domain (set of possible pixel values) respectively. G_{σ_s} is a two-dimensional spatial Gaussian kernel while G_{σ_r} is a range Gaussian kernel where σ_s and σ_r are the spatial and range filtering parameters for the image I. The kernel can be represented as:

$$G_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{13}$$

Thus, $G_{\sigma_s}(\|p-q\|)$, which defines the spatial distance, decreases influence of spatially distant pixels compared to the position p. The parameter σ defines the extension of the neighborhood. Similarly, $G_{\sigma_r}\left(I_p - I_q\right)$ decreases influence of q pixels with color intensities different from that of p (I_p).

The range parameter σ_r differentiates bilateral filter from Gaussian filter which only takes into consideration spatial (location) closeness for smoothing. As σ_r increases, the bilateral filter converges to a Gaussian blur filter. As σ_s increases, larger features are smoothed. Both spatial and range weights are multiplied in bilateral filter; thus, no smoothing occurs even if one weight approaches zero. For example, a narrow range Gaussian combined with large spatial Gaussian will produce limited smoothing in spite of large spatial extent. The contours are maintained by the range weight. This will help the bilateral filter to remove noisy thresholds from the heatmap while maintaining the sharp edges formed by recurrent congestion patches. Please refer to Paris et al. (2007) for a detailed review.

3.3.2. Total Variation based Threshold Denoising

Total variation (TV), first proposed by Rudin et al. (1992), has been used extensively for image denoising problems since they are known to denoise the image without smoothing the object boundaries. Many algorithms have been developed in the past for total variation based image denoising. Rodríguez (2013) provides a detailed review of the different methods developed for TV-denoising. In this study, we adopted the algorithm proposed by Chambolle (2004) for solving the minimization problem of total variation of an image. As explained in Section 3.3.1, the image can be expressed as a second-order tensor, given by $I : \tau \in \mathbb{R}_{+}^{n\times N'}$. Denoting $X = \mathbb{R}_{+}^{n\times N'}$ and $Y = X \times X$, for any $u \in X$, the gradient $\nabla u \in Y$ is given by

$$(\nabla u)_{i,j} = \left((\nabla u)^x_{i,j}, (\nabla u)^y_{i,j}\right) \tag{14}$$

with

$$(\nabla u)^x_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} & \text{if } i < n, \\ 0 & \text{if } i = n, \end{cases} \tag{15}$$

$$(\nabla u)^y_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} & \text{if } j < N', \\ 0 & \text{if } j = N', \end{cases} \tag{16}$$

Then, the discretized total variation can be defined as

$$J(u) = \sum_{\substack{1\le i\le n,\\ 1\le j\le N'}} \left|(\nabla u)_{i,j}\right|_Y \tag{17}$$

where the standard Euclidean scalar product is given by

$$\langle p, q\rangle_Y = \sum_{\substack{1\le i\le n,\\ 1\le j\le N'}} p^1_{i,j} q^1_{i,j} + p^2_{i,j} q^2_{i,j}$$

for all $p, q \in Y$, $p = \left(p^1, p^2\right)$ and $q = \left(q^1, q^2\right)$. Then, Equation 17 can be written as

$$J(u) = \sup\left\{\langle p, \nabla u\rangle_Y : p \in Y, \left|p_{i,j}\right| \le 1 \forall i, j\right\} \tag{18}$$

Now, a discrete divergence operator div:$Y \to X$ can be introduced $\forall p \in Y$ and $\forall u \in X$ defined by,

$$\langle -\mathrm{div} p, \nabla u\rangle_X = \langle p, \nabla u\rangle_Y$$

It can be easily shown that the div is given by

$$(\mathrm{div} p)_{i,j} = \begin{cases} p^1_{i,j} - p^1_{i-1,j} & \text{if } 1 < i < n, \\ p^1_{i,j} & \text{if } i = 1, \\ -p^1_{i-1,j} & \text{if } i = n, \end{cases} + \begin{cases} p^2_{i,j} - p^2_{i,j-1} & \text{if } 1 < j < N', \\ p^2_{i,j} & \text{if } j = 1, \\ -p^1_{i,j-1} & \text{if } i = N', \end{cases} \tag{19}$$

From the definition of divergence operator and Equation 18, we can write

$$J(u) = \sup_{v\in K_F} \langle u, v\rangle_X$$

where K_F is given by

$$\left\{\mathrm{div} p :: p \in Y, \left|p_{i,j}\right| \le 1 \forall i, j\right\}$$

It can be shown that determination of nonlinear projection $\pi_{\frac{1}{\lambda}K_F}(f)$ leads to solving the following constrained minimization problem (Chambolle, 2004; Duran et al.).

$$\min\left\{\|\lambda \mathrm{div} p - f\|_X^2 : p \in Y, \left|p_{i,j}\right|^2 - 1 \leq 0, \forall i, j\right\} \tag{20}$$

From Karush-Kuhn-Tucker optimality conditions (Hiriart-Urruty and Lemaréchal, 1993; Ciarlet, 1982), we get:

$$-\nabla(\lambda \mathrm{div} p - f)_{i,j} + \alpha_{i,j} p_{i,j} = 0 \tag{21}$$

where either $\left|p_{i,j}\right| < 1$, $\alpha_{i,j} p_{i,j} = 0$, and $\nabla(\lambda \mathrm{div} p - f)_{i,j} = 0$; or $\left|p_{i,j}\right| = 1$ and $\alpha_{i,j} p_{i,j} > 0$. Thus,

$$\alpha_{i,j} p_{i,j} = \left|\nabla(\lambda \mathrm{div} p - f)_{i,j}\right|$$

Chambolle proposed a semi-implicit gradient descent algorithm for solving the minimization problem. For denoising parameter λ, tolerance parameter t, time-step τ, and while $\max_{\forall i,j}\left\{\left|p_{i,j}^{n+1} - p_{i,j}^{n}\right|\right\} > t$,

$$p_{i,j}^{n+1} = \frac{p_{i,j}^{n} + \tau(\nabla(\mathrm{div} p^n - f/\lambda))_{i,j}}{1 + \tau\left|(\nabla(\mathrm{div} p^n - f/\lambda))_{i,j}\right|} \tag{22}$$

Chambolle (2004) showed that the algorithm converges for $\tau \leq 1/8$. In this study, we chose $t = 0.0002$ and our objective is to determine the optimal denoising parameter λ. Interested readers can also refer to Duran et al. for a detailed description of Chambolle's algorithm.

3.4. Overall AID framework

The flowchart of the proposed AID algorithm framework is shown in Figure 2. Historical traffic data are processed weekly in MapReduce for univariate speed thresholds computation of 15-minute windows for each segment and day-of-week. The speed thresholds can be generated using SND, MAD or IQD method. Next, the thresholds are used to generate speed threshold heatmaps for each road and direction. These heatmaps are denoised using total variation or bilateral filter. The thresholds are matched with real-time speed data and when speed is less than threshold for 3 consecutive intervals, incident alarm is triggered. This persistence test is performed to reduce false alarms generated due to spurious noise in real-time data, similar to the study of Li et al. (2013).

3.5. Performance Measures

Incident detection performance has been evaluated in terms of the performance measures used in past studies (Parkany and Xie, 2005; Ren et al., 2012; Li et al., 2013). This involves determination of 4 performance measures:

1. Detection Rate (DR) is the ratio of number of incidents detected by AID algorithm to the total number of incidents occurred.

$$DR = \frac{\text{Total number of detected incidents}}{\text{Total number of actual incidents}} \times 100\% \tag{23}$$

2. False Alarm Rate (FAR) is used for penalizing false calls and is defined as the ratio of number of false alarms reported to total number of AID algorithm application. For example, if traffic data is reported at one-minute interval for a particular segment and AID algorithm is applied for each record, then the number of algorithm application is equal to 60. Thus, if 5 records out of them is reported as false calls, then FAR is equal to $[(5/60) \times 100]\,\% = 8.3\%$

$$FAR = \frac{\text{Total number of false alarm cases}}{\text{Total number of algorithm applications}} \times 100\% \tag{24}$$

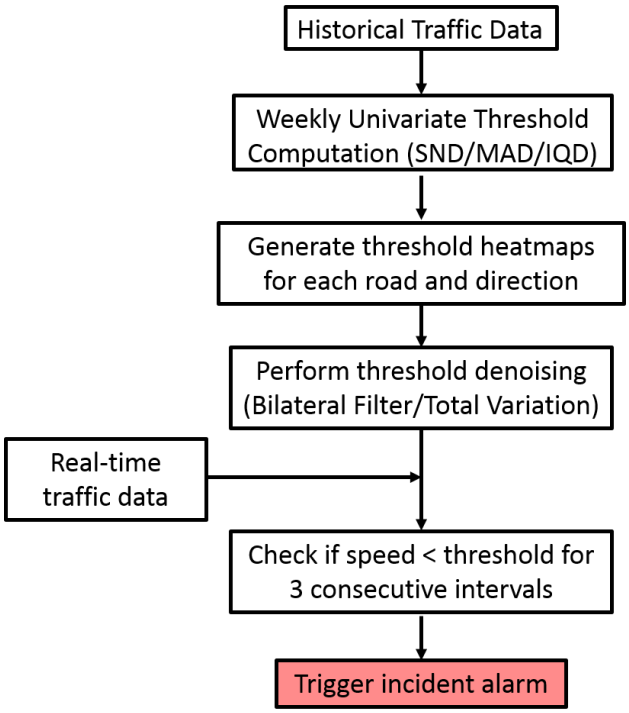


Figure 2: Illustration of the AID algorithm framework

3. Mean Time to Detect ($MTTD$) takes into consideration the latency involved in the AID algorithm. It is defined as the average of time elapsed between actual start of the incident and time when the incident is first detected by the algorithm.

$$MTTD = \frac{\text{Total time elapsed between detecting incidents}}{\text{Total number of incidents detected}} \tag{25}$$

4. Performance Index (PI), given by Equation 26, brings together all 3 performance measures (DR, FAR, and $MTTD$) into a single measure to find out the overall performance of the AID algorithm. PI is believed to be one of the best possible measure to reflect the performance of AID during model selection (Ren et al., 2012). Minimizing PI is the optimization objective used during cross-validation. Since DR can be 100% or FAR can be 0% during training, the PI measure is slightly modified with the constants (1.01 and 0.001) to handle such cases, similar to (Ren et al., 2012).

$$PI = \left(1.01 - \frac{DR}{100}\right) \times \left(\frac{FAR}{100} + 0.001\right) \times MTTD \tag{26}$$

4. Data Description

The data used in the study comprises traffic speed data and crash data from Interstate Freeways I-80/35 and I-235 of the Des Moines region, in Iowa, USA. The Des Moines region experiences the majority of Iowa's freeway congestion (62% of slow traffic events) along with one of the highest concentration of traffic incidents within the state (Kapsch, 2016). Hence, it is a challenging task to separate traffic incidents from recurring congestion events in such a network and develop a reliable AID framework.

The study period extended from April, 2017 to October, 2017. A total of 210 lane-blocking traffic incidents were reported during this period in the study region. The incident reports were obtained from the Traffic Management Center records in Ankeny, Iowa. The incident reports included information on the location of the incident (latitude,

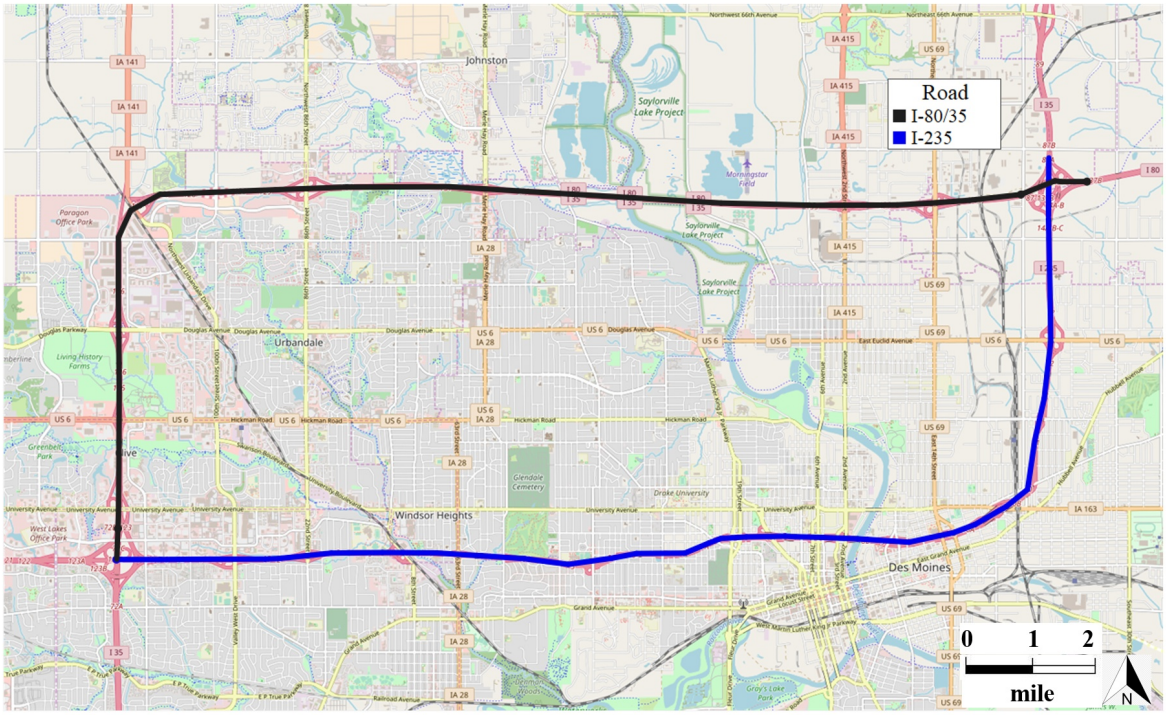


Figure 3: Study region in Des Moines, Iowa

longitude, road, direction), start and end-time of the incidents, and also the incident type. The following incident classes were included in this study: 1-vehicle crash, 2-vehicle crash, 3+ vehicle crash, stalled vehicle, and debris. Construction reports and slow traffic events (not associated with any traffic incidents) were excluded from the incident database. All the incidents were manually verified with the cameras installed in the freeways.

High-resolution probe-based traffic speed data, provided by INRIX (INRIX, 2018) is used in this study. The entire road network is divided into approximately 0.5 miles long segments and average speed data is reported in 1-minute interval. Since quality of probe-based speed data depends on the number of probe vehicles available, INRIX reports two parameters, confidence score and c-value, to denote the reliability of each traffic record. Confidence score can take 3 values: 10, 20, and 30. Confidence score of 30 indicates that only real-time probe vehicles are used for reporting real-time speed. On the other hand, confidence score 10 indicates historical traffic speed data and is used to report traffic speed of a segment due to unavailability of probe vehicles. When a mix of real-time probe data and historical speed data is used, then confidence score of 20 is reported. C-value is an additional reliability parameter provided by INRIX only when confidence score is 30. C-value can range from 0-100 and provides a relative measurement of number of probe-vehicles used for real-time speed report. Since traffic incident detection requires real-time speed reports, we used traffic speed data corresponding to confidence score of 30 and c-value greater than 30, as suggested by Haghani et al. (2009). Since the study region has considerably high traffic volume (annual average daily traffic of 70,000 vehicles in 2017), hence more than 98% of the traffic speed records used in this study were real-time.

5. Results

The first step for incident detection is the determination of speed thresholds from the previous 8 weeks of historical traffic data. This involves determination of the optimum threshold constant c for each the 3 methods, SND, MAD, and IQD (see Equations 5, 6, and 7 respectively). Figure 4 shows the variation of the different performance measures (DR, FAR, $MTTD$, and PI) for different values of the threshold constant (c) for the 3 AID methods (SND, MAD, and IQD). We also experimented with 2 weeks, 4 weeks, and 12 weeks of historical data for threshold computation. However, 8 weeks historical data were found to be optimum and hence used throughout this study.

Figure 4a shows that while MAD has the least variation of DR with increasing values of c, DR decreases sharply for SND for higher values of c. On the other hand, FAR remains fairly high for all values of c for MAD while the least FAR is achieved for SND. This behavior is atypical in machine learning community where better DR comes at the cost of higher FAR. IQD tries to achieve "best of the two worlds" where DR remains significantly high without producing too many false calls. $MTTD$ remains fairly constant for all 3 methods for higher values of c except for SND where $MTTD$ increases for higher c values. The combined performance with all 3 measures is depicted by PI in Figure 4d. It shows that SND being sensitive to outliers or anomalies is very sensitive to the threshold constant c. The minimum PI is obtained for IQD with $c_{IQD} = 2$. In the rest of this study, the optimum c_{IQD} is used for threshold determination. Speed thresholds for a sample segment with optimum c value for each of the 3 methods is shown next to discuss the advantages and disadvantages of each method.

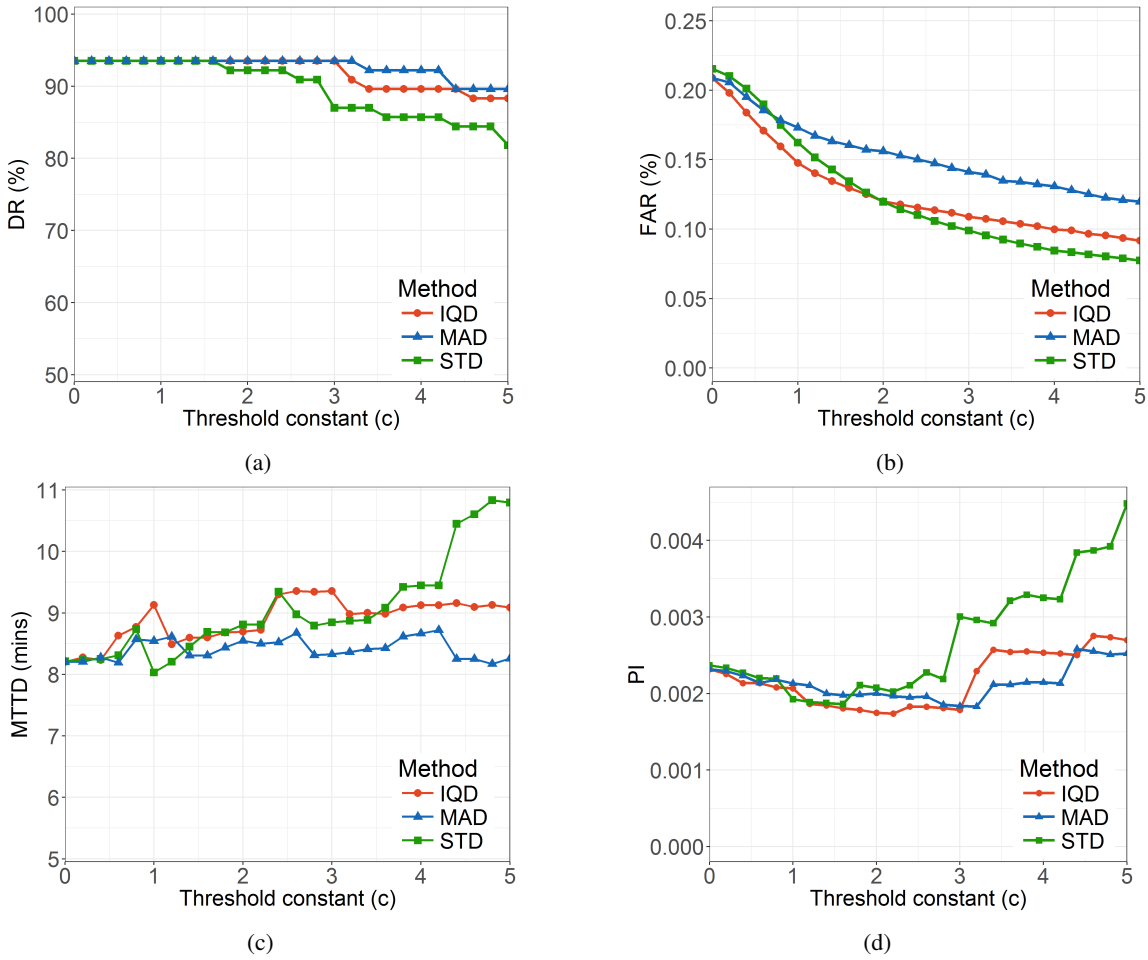


Figure 4: Variation of (a) DR, (b) FAR, (c) MTTD, and (d) PI for different threshold constants c for SND, MAD, and IQD methods

Figure 5 shows the speed threshold values for a particular segment of I-235 E for a given weekday. The recurring morning peak congestion resulted in low threshold values between 7 AM to 9 AM as shown in Callout (ii). However, MAD has higher speed thresholds ($\geq$ 30 mph) compared to both IQD and SND (20 mph to 30 mph threshold values). Evening peak, shown in Callout (iii), has lower effect on traffic compared to morning peak primarily because evening peak affects mostly opposite direction of traffic (I-235 W). Nonetheless, speed thresholds are still highest for MAD and lowest for SND in evening peak time too. IQD follows closely to the SND thresholds during peak times. However, night-time construction during 1 out of 8 weeks of historical data resulted in low threshold values during night-time for SND, as shown in Callouts (i) and (iv). IQD and MAD, being robust to outliers, are not affected by such sporadic

night-time congestion due to construction purposes and produces high speed thresholds. This means that incident alarm will be triggered during night-time with IQD and MAD methods if congestion occurs due to any incident (which requires attention of traffic incident managers) or due to some night-time construction (which traffic incident managers are already aware of from construction schedule).

SND, on other hand, might miss such an incident due to lower threshold values. Similarly, low thresholds by SND during peak times will result in lower incident detection rates along with low false alarm rates. On the other hand, MAD having high threshold values even during peak times will lead to higher detection rates at the expense of high false alarm rates too. IQD attempts to achieve the "best of the two worlds" with low threshold values during peak times while maintaining high thresholds all other times. This is further demonstrated in Figure 4 where SND has lowest *DR* along with the lowest *FAR* while MAD has highest *DR* along with highest *FAR*. IQD, on the other hand, has significantly high *DR* but with not too high *FAR*. Thus, IQD can be used a significantly better alternative than the popular SND algorithm for obtaining high incident detection rates without compromising too many false alarms.

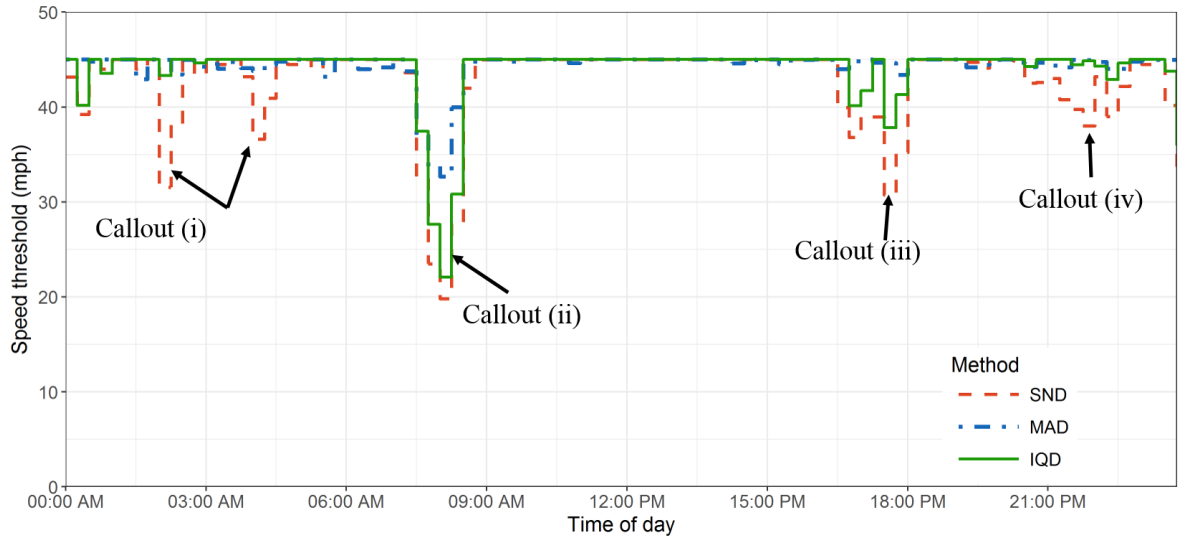
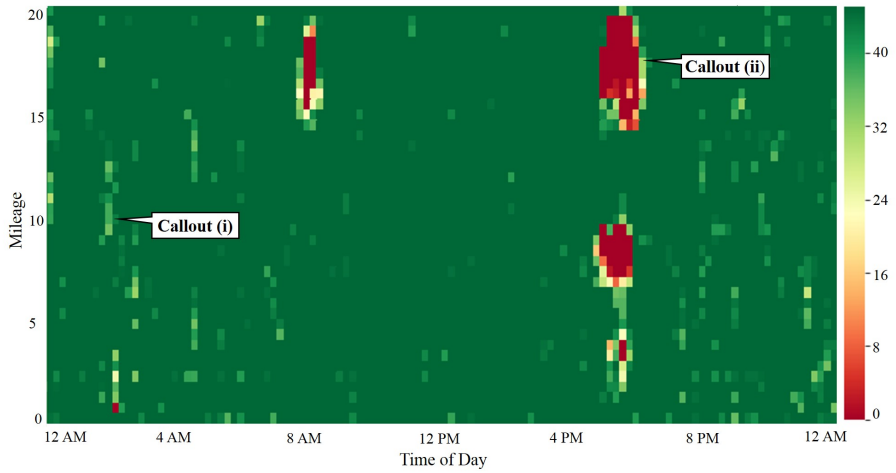


Figure 5: Sample speed thresholds of SND, MAD, and IQD for a particular segment in a given weekday

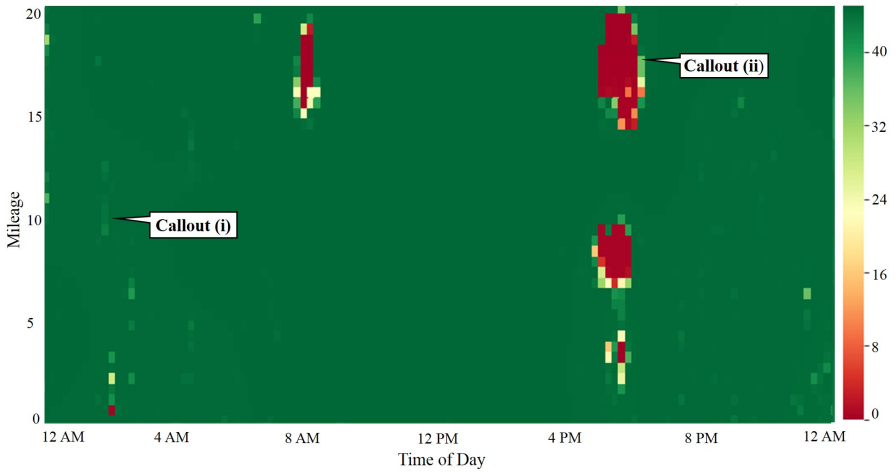
IQD and other univariate threshold computation methods (e.g., SND and MAD) do not take into consideration the spatio-temporal correlations of the 15-minute threshold windows. Thus, thresholds computed can be noisy and can be improved by considering the topology of the network. In this study, we considered the speed thresholds for all segments in a particular direction for a given day and week as a "speed heatmap" and performed two image denoising techniques, bilateral filter and total variation, to obtain smoother accurate threshold information. Figure 6a shows a sample raw threshold speed heatmap, the denoised one with bilateral filter (6b), and also with total variation (6c).

It can be seen that the denoising techniques helps to remove the intermittent or sporadic low threshold values, shown in 'Callout (i)', generated typically during night times or non-peak hours either due to some outlying observations caused by incidents or other data issues. On the other hand, the threshold speed values at the edges of peak time congestion (as shown in 'Callout (ii)') gets increased slightly with total variation (6c) denoising compared to the raw threshold heatmap (6a). However, bilateral filter maintains the edges intact since it considers both the spatial and color range kernels for smoothing purposes. This attribute is important for incident detection because increase in threshold values at edges of peak hours might result in missing incidents occurring during those times, which in turn will result in lowered incident detection rates.

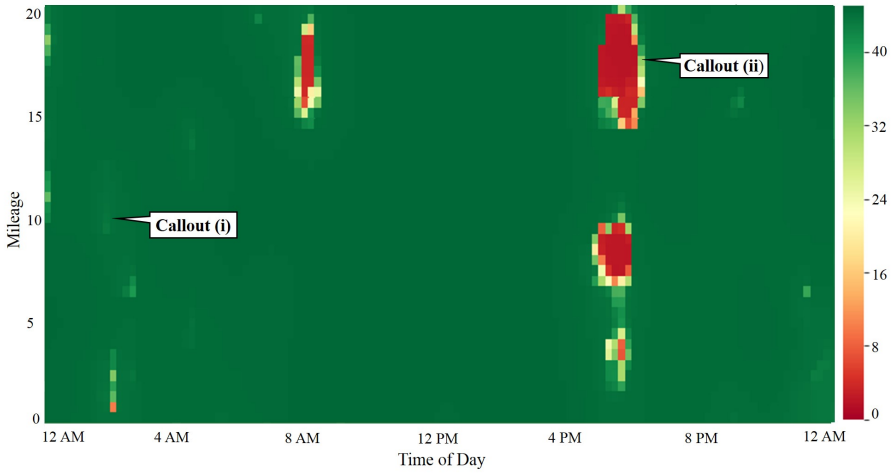
Incident detection performances will vary depending on the amount of denoising. Hence, our next objective is to find out the optimal hyperparameters for denoising. The two hyperparameters in bilateral filter denoising are σ_s and σ_r, given in Equation 12. Optimal σ_r usually depends on the amount of noise in the image and estimated in terms of σ_r/σ_n(Zhang and Gunturk, 2008), where σ_n denotes the noise standard deviation. Since σ_n is unknown for the speed threshold heatmaps, we used the ratio σ_r/σ_i for hyper-parameter selection where σ_i denotes the standard deviation of



(a)



(b)



(c)

Figure 6: Speed threshold heatmap in mph (a) without denoising, (b) bilateral filter denoising, and (c) total variation denoising

image. We assume here that the noise in the heatmap is proportional to color range in the heatmap. Figure 7 shows the contour plot of the variation of DR, FAR, $MTTD$, and PI with σ_s and σ_r/σ_i. Figure 7a shows that DR is improved from 91.5% to 96% with increase in σ_s and $\sigma_r/\sigma_i > 1$. Although FAR also increases with denoising, however the increment is only from 0.12% to 0.136%. $MTTD$ is found to increase from 8.4 minutes to 9.1 minutes from higher σ_s and σ_r/σ_i, except for very high σ_s and σ_r/σ_i ($\sigma_s > 6$ and $\sigma_r/\sigma_i > 2$). Although FAR and $MTTD$ are found to increase amrginally with increase in denoising, however the overall performance index is found to improve (lower PI) with denoising the speed threshold heatmaps, as shown in Figure 7d.

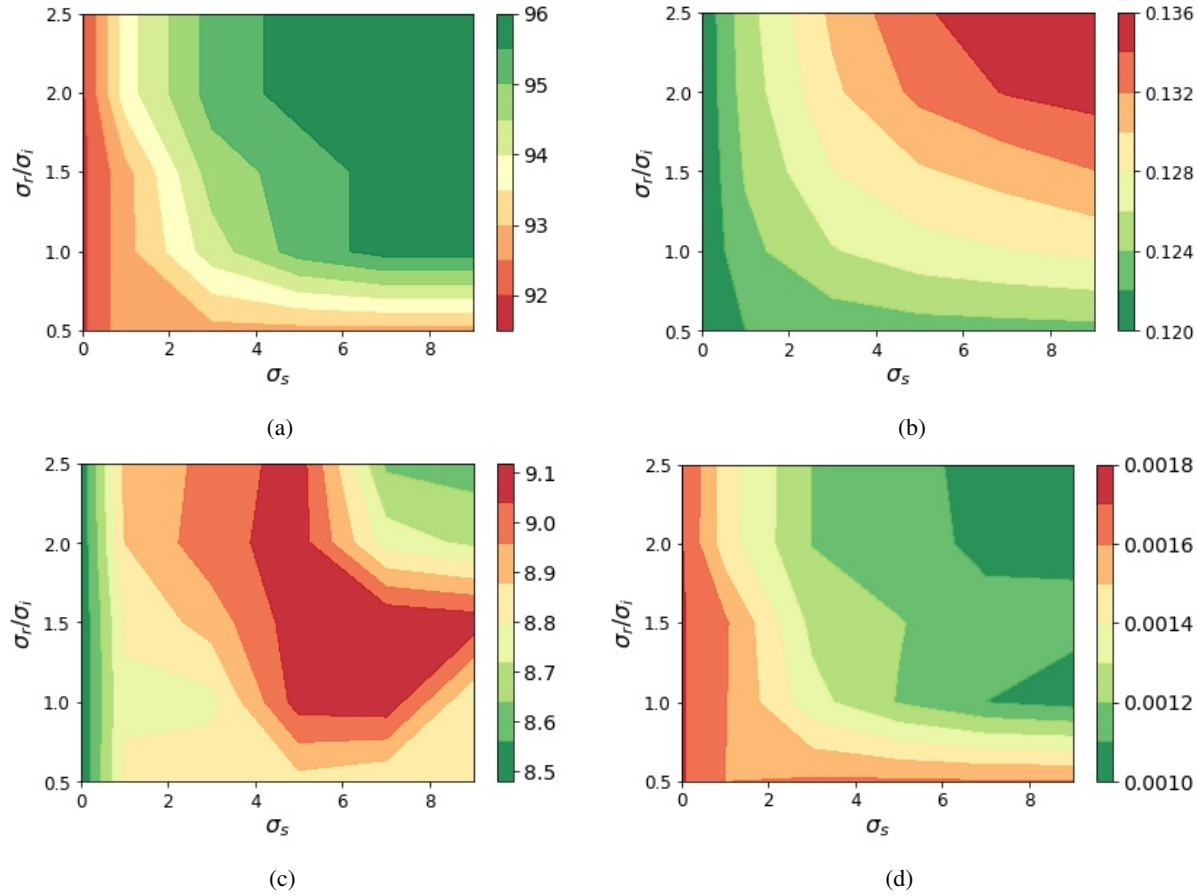


Figure 7: Contour plots showing variation of (a) DR (%), (b) FAR(%), (c) MTTD (mins), and (d) PI with σ_s and σ_r/σ_i

Total variation denoising depends on the optimal hyper-parameter λ (see Equation 22). To compare the performance of total variation (TV) with bilateral filter (BL) denoising simultaneously, Figure 8 shows the variation of DR, FAR, $MTTD$, and PI w.r.t λ for TV and σ_s for BL. Since BL has two hyper-parameters (σ_s and σ_r), for each σ_s, the σ_r/σ_i producing optimal PI is used. Figure 8a shows that although DR improves with denoising for BL, however it is found to decrease for TV denoising. On the other hand, while FAR and $MTTD$ remains fairly constant with TV denoising, they are found to increase slightly with BL denoising (Figures 8b and 8b). Overall, PI improvement (i.e., decrease in PI) is only obtained for BL denosing while for TV, PI is found to increase primarily due to lowering of DR with increased TV denoising. Thus, bilateral filter (BL) denoising for speed threshold heatmaps helps to achieve improved incident detection performances while total variation (TV) denoising fails to achieve such improvement. To further investigate the reason for such behavior, we plotted the change in DR and FAR over the time of the day with increase in denoising by BL and TV.

To find out the effect of denoising of threshold heatmaps on DR and FAR, the hourly improvement in DR and FAR with respect to raw heatmaps are shown in Figures 9 and 10 respectively. The DR and FAR change is obtained

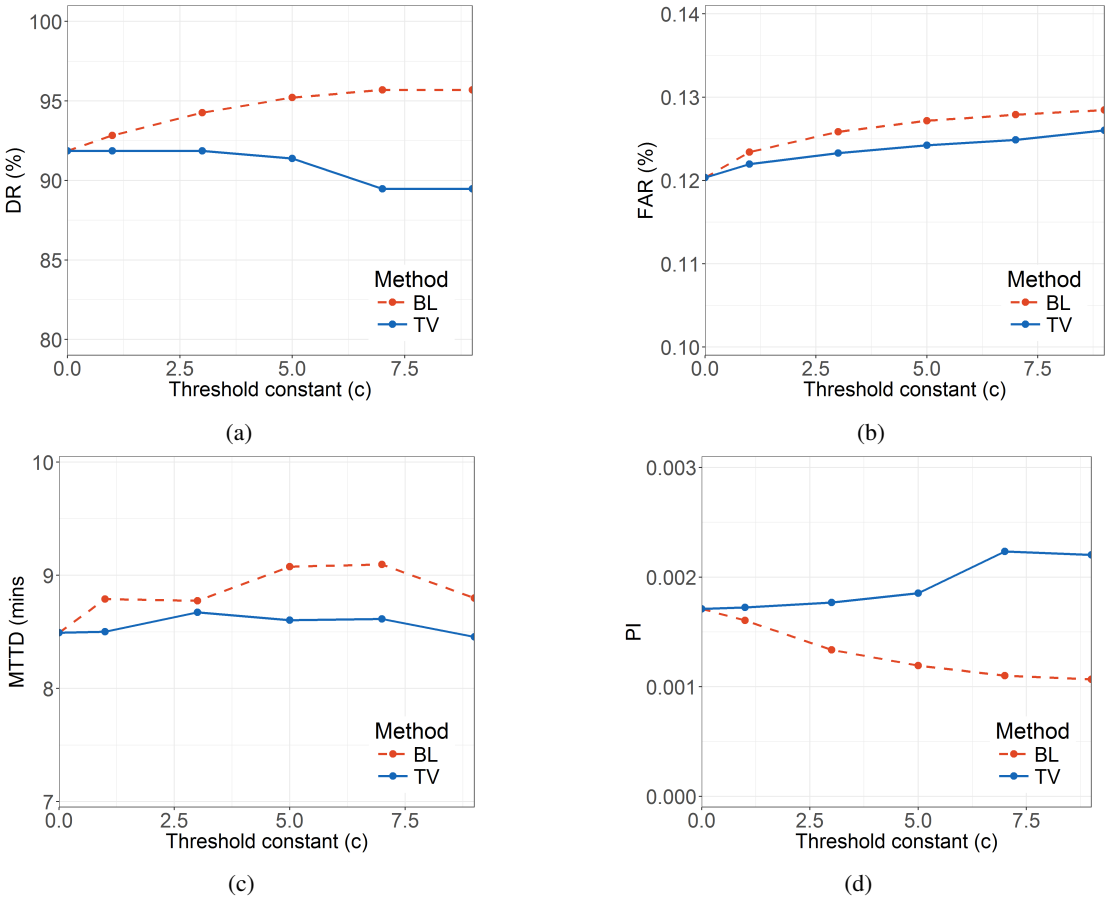
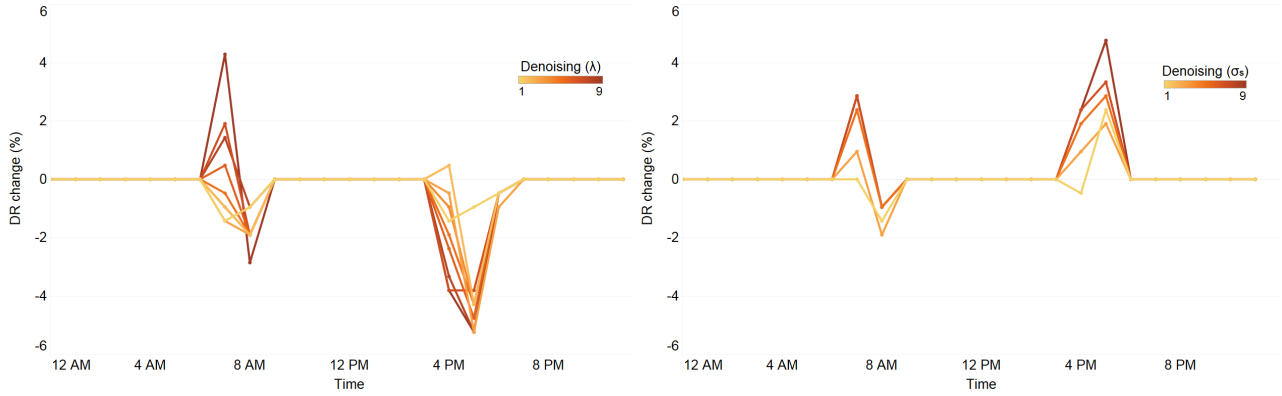


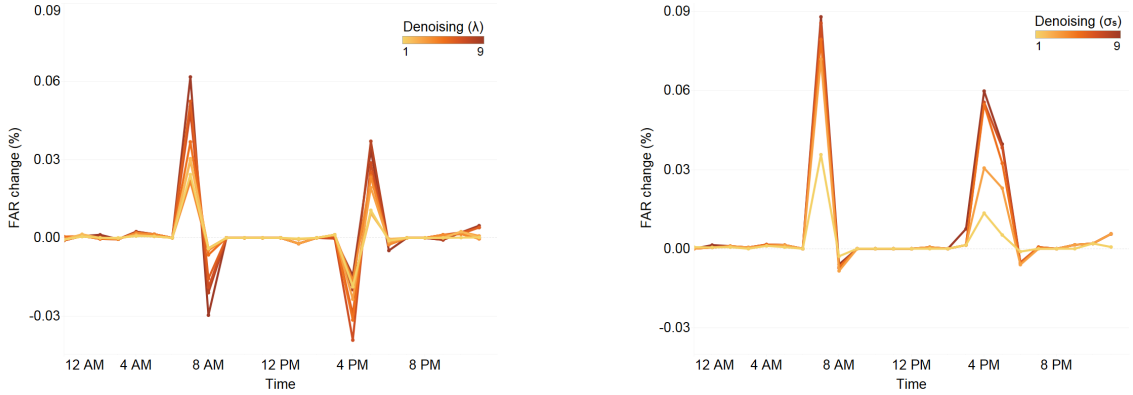
Figure 8: Variation of (a) DR (%), (b) FAR(%), (c) MTTD (mins), and (d) PI with denoising parameter, σ_s for bilateral filter and λ for total variation

by subtracting hourly *DR* and *FAR* obtained using denoised heatmaps to those obtained using raw heatmaps. Figure 9a shows that although total variation denoising with $\lambda > 5$ helps in improvement of *DR* during morning peak hours (7 AM), however *DR* decreases during evening peak hours (4 PM and 5 PM) for all values of λ. This can be attributed to the increase in threshold speed values due to denoising by total variation in evening peak hours, shown in Callout (ii) of Figure 6. On the other hand, bilateral filter denoising helps in improving *DR* in both morning and evening peak hours since it doesn't affect the low threshold values due to the color range parameter (σ_r). As expected, increase in *DR* also comes at the cost of increase in *FAR* and vice versa as shown in Figure 10. For total variation denoising, *FAR* increases during morning peak hours and decreases during transition from morning peak to normal conditions. During evening peak hours, although *FAR* decreases in the beginning (4 PM), however, it increases again during the later phase of evening peak hours. As an end result, *FAR* remains almost constant with total variation denoising as seen in Figure 8b. On the other hand, *FAR* increase due to bilateral filter denoising, which is expected given the fact that *DR* increased with denoising. Thus, it can be stated that the primary reason for reduction of *DR* and in turn, reduction of *PI* for total variation denoising is primarily due to missing of detected incidents during peak hours congestion where speed threshold increases due to total variation denoising. Bilateral filter, on the other hand, maintains low threshold values intact even during peak hours with removal of spurious noisy thresholds and achieves better detection performance (higher *DR* and *PI*) compared to raw threshold values generated by univariate threshold computation (*IQD*).



(a) (b)

Figure 9: Hourly change in *DR* due to speed threshold denoising by (a) total variation and (b) bilateral filter



(a) (b)

Figure 10: Hourly change in *FAR* due to speed threshold denoising by (a) total variation and (b) bilateral filter

6. Conclusions

Automatic traffic incident detection has been identified to be crucial for reduction of non-recurrent congestion caused by incidents. In this study, we proposed an AID framework can be implemented over large traffic networks due to the inherent parallel computation involved in the framework. The proposed methodology consists of two major improvements to the AID algorithms developed in past literature. The first step of our AID framework consists of estimation of robust summary statistics (speed thresholds) across non-overlapping windows of time series data produced from each road segment. This dimension reduction step can be parallelized using MapReduce framework making it feasible to apply the algorithm over large traffic networks. We then visualize these summary statistics as threshold heatmaps leveraging the spatio-temporal correlations of the time windows and perform multivariate denoising of the heatmaps. Such denoised thresholds can produce more accurate representations of the normal traffic patterns which in turn can help in increasing incident detection performance.

To produce robust summary statistics for the non-overlapping time windows of each time series traffic data, we proposed MAD and IQD methods in place of the popular SND algorithm. In these methods, we replace the mean and standard deviation measures which are known to be prone to outliers with the comparatively robust statistics such as median, MAD, and IQD. SND being prone to outliers is found to be affected by occasional low speeds caused due to incidents or traffic constructions. This results in lowering speed thresholds which in turn produces low incident detection rates while having false alarm rates comparable to IQD. MAD, on the other hand, is found to produce

high threshold values even during peak hours thereby resulting in significantly higher false alarm rates than SND or IQD. Our results show that the IQD method, which uses median and IQD as summary statistics, achieves the best performance overall in terms of incident detection. It can achieve high incident detection rates without increasing false alarm rates significantly.

In our next step, we construct heatmaps with the IQD thresholds and perform image denoising of the heatmaps. We used two edge-preserving image denoising techniques, bilateral filter and total variation. Our results show that speed thresholds increases in the recurrent congestion patches due to total variation denoising. As a result, detection rate decreases and the overall performance, given by PI, deteriorates. Bilateral filter, on the other hand, preserves the edges and the low thresholds in recurrent congestion patches intact and simultaneously removes spurious noisy thresholds. This helps in achieving higher detection rates with bilateral filter denoising without increasing false alarms significantly thereby improving overall performance of the AID algorithm.

The entire framework is data-driven in nature and can be easily extended in other data-set or traffic networks. However, there is also scope of further improvements on this AID framework. This study assumed uniform weights for the road segments while constructing threshold heatmaps. In future, it can be improved by using weights based on the road segment length or finding out the optimal weights using data-driven methods. The denoising framework can also be parallelized thereby unifying the entire system setup. And other advanced image denoising techniques can be applied to obtain better estimates of the thresholds from heatmaps.

Acknowledgements

Our research results are based upon work jointly supported by the National Science Foundation Partnerships for Innovation: Building Innovation Capacity (PFI: BIC) program under Grant No. 1632116, National Science Foundation under Grants No. CNS-1464279, CCF-1566281, CCF-1750920, Award ATD-1830254 supported by NSF and the National Geospatial-Intelligence Agency, and Iowa DOT Office of Traffic Operations Support Grant. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Aggarwal, C.C., 2007. Data streams: models and algorithms. volume 31. Springer Science & Business Media. doi:10.1007/978-0-387-47534-9.

Anbaroglu, B., Heydecker, B., Cheng, T., 2014. Spatio-temporal clustering for non-recurrent traffic congestion detection on urban road networks. Transportation Research Part C: Emerging Technologies 48, 47–65. doi:10.1016/j.trc.2014.08.002.

Apache Pig, 2018. https://pig.apache.org/. Accessed July 20, 2018.

Asakura, Y., Kusakabe, T., Nguyen, L.X., Ushiki, T., 2017. Incident detection methods using probe vehicles with on-board gps equipment. Transportation Research Rart C: Emerging Technologies 81, 330–341. doi:10.1016/j.trc.2016.11.023.

Balke, K., Dudek, C., Mountain, C., 1996. Using probe-measured travel times to detect major freeway incidents in Houston, Texas. Transportation Research Record: Journal of the Transportation Research Board 1554, 213–220. doi:10.3141/1554-25.

Chakraborty, P., Hegde, C., Sharma, A., 2017a. Trend filtering in network time series with applications to traffic incident detection, in: Time Series Workshop, 31st Conference on Neural Information Processing Systems (NIPS).

Chakraborty, P., Hess, J.R., Sharma, A., Knickerbocker, S., 2017b. Outlier mining based traffic incident detection using big data analytics, in: Transportation Research Board 96th Annual Meeting Compendium of Papers, pp. 8–12.

Chambolle, A., 2004. An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision 20, 89–97. doi:10.1023/B:JMIV.0000011325.36760.1e.

Chen, Z., Liu, X.C., Zhang, G., 2016. Non-recurrent congestion analysis using data-driven spatiotemporal approach for information construction. Transportation Research Part C: Emerging Technologies 71, 19–31. doi:10.1016/j.trc.2016.07.002.

Chung, Y., 2011. Quantification of nonrecurrent congestion delay caused by freeway accidents and analysis of causal factors. Transportation Research Record: Journal of the Transportation Research Board 2229, 8–18. doi:10.3141/2229-02.

Chung, Y., Recker, W.W., 2012. A methodological approach for estimating temporal and spatial extent of delays caused by freeway accidents. IEEE Transactions on Intelligent Transportation Systems 13, 1454–1461. doi:10.1109/TITS.2012.2190282.

Ciarlet, P.G., 1982. Introduction à l'analyse numérique matricielle et à l'optimisation. Dunod.

Dean, J., Ghemawat, S., 2008. Mapreduce: simplified data processing on large clusters. Communications of the ACM 51, 107–113. doi:10.1145/1327452.1327492.

Dowling, R., Skabardonis, A., Carroll, M., Wang, Z., 2004. Methodology for measuring recurrent and nonrecurrent traffic congestion. Transportation Research Record: Journal of the Transportation Research Board 1867, 60–68. doi:10.3141/1867-08.

Dudek, C.L., Messer, C.J., Nuckles, N.B., 1974. Incident detection on urban freeways. Transportation Research Record 495, 12–24.

Duran, J., Coll, B., Sbert, C., . Chambolle's projection algorithm for total variation denoising. Image processing On Line 2013, 311–331. doi:10.5201/ipol.2013.61.

FHWA, 2005. Travel time reliability: Making it there on time, all the time. https://ops.fhwa.dot.gov/publications/tt_reliability/TTR_Report.htm.

Gupta, M., Gao, J., Aggarwal, C.C., Han, J., 2014. Outlier detection for temporal data: A survey. IEEE Transactions on Knowledge and Data Engineering 26, 2250–2267. doi:10.1109/TKDE.2013.184.

Haghani, A., Hamedi, M., Sadabadi, K.F., 2009. I-95 corridor coalition vehicle probe project: Validation of INRIX data. Technical Report. I-95 Corridor Coalition.

Hampel, F.R., 1974. The influence curve and its role in robust estimation. Journal of the American Statistical Association 69, 383–393. doi:10.2307/2285666.

Hellinga, B., Knapp, G., 2000. Automatic vehicle identification technology-based freeway incident detection. Transportation Research Record: Journal of the Transportation Research Board 1727, 142–153. doi:10.3141/1727-18.

Hiriart-Urruty, J.B., Lemaréchal, C., 1993. Convex analysis and minimization algorithms I, II. volume 305-306. Springer Science & Business Media.

INRIX, 2018. http://inrix.com/. Accessed July 20, 2018.

Kamran, S., Haas, O., 2007. A multilevel traffic incidents detection approach: Identifying traffic patterns and vehicle behaviours using real-time GPS data, in: 2007 Intelligent Vehicles Symposium, IEEE. pp. 912–917. doi:10.1109/IVS.2007.4290233.

Kapsch, 2016. Traffic management center annual report 2016. Iowa Department of Transportation.

Leys, C., Ley, C., Klein, O., Bernard, P., Licata, L., 2013. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. Journal of Experimental Social Psychology 49, 764–766. doi:10.1016/j.jesp.2013.03.013.

Li, X., Lam, W.H., Tam, M.L., 2013. New automatic incident detection algorithm based on traffic data collected for journey time estimation. Journal of Transportation Engineering 139, 840–847. doi:10.1061/(ASCE)TE.1943-5436.0000566.

Li, Y., McDonald, M., 2005. Motorway incident detection using probe vehicles, pp. 11–15. doi:10.1680/tran.2005.158.1.11.

Noland, R.B., Polak, J.W., 2002. Travel time variability: A review of theoretical and empirical issues. Transport Reviews 22, 39–54. doi:10.1080/01441640010022456.

Owens, N., Armstrong, A., Sullivan, P., Mitchell, C., Newton, D., Brewster, R., Trego, T., 2010. Traffic incident management handbook. Technical Report. U.S. Department of Transportation.

Ozbay, K., Kachroo, P., 1999. Incident management in intelligent transportation systems. Artech House Publishers.

Paris, S., Kornprobst, P., Tumblin, J., Durand, F., 2007. A gentle introduction to bilateral filtering and its applications, in: ACM SIGGRAPH 2007 courses, ACM. doi:10.1145/1281500.1281602.

Parkany, E., Bernstein, D., 1995. Design of incident detection algorithms using vehicle-to-roadside communication sensors. Transportation Research Record: Journal of the Transportation Research Board 1494, 67–74.

Parkany, E., Xie, C., 2005. A complete review of incident detection algorithms & their deployment: what works and what doesn't. Technical Report NETCR 37. The New England Transportation Consortium.

Pearson, R.K., 2005. Mining imperfect data: Dealing with contamination and incomplete records. SIAM. doi:10.1137/1.9780898717884.

Ren, J.S., Wang, W., Wang, J., Liao, S., 2012. An unsupervised feature learning approach to improve automatic incident detection, in: 2012 15th International IEEE Conference on Intelligent Transportation Systems, IEEE. pp. 172–177. doi:10.1109/ITSC.2012.6338621.

Rodríguez, P., 2013. Total variation regularization algorithms for images corrupted with different noise models: A review. Journal of Electrical and Computer Engineering 2013, 1–18. doi:10.1155/2013/217021.

Rudin, L.I., Osher, S., Fatemi, E., 1992. Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60, 259–268. doi:10.1016/0167-2789(92)90242-F.

Schrank, D.L., Lomax, T.J., 2007. The 2007 urban mobility report. Technical Report. Texas Transportation Institute, The Texas A&M University.

Sethi, V., Bhandari, N., Koppelman, F.S., Schofer, J.L., 1995. Arterial incident detection using fixed detector and probe vehicle data. Transportation Research Part C: Emerging Technologies 3, 99–112. doi:10.1016/0968-090X(94)00017-Y.

Snelder, M., Bakri, T., van Arem, B., 2013. Delays caused by incidents: Data-driven approach. Transportation Research Record: Journal of the Transportation Research Board 2333, 1–8. doi:10.3141/2333-01.

Sussman, J.M., 2005. Perspectives on Intelligent Transportation Systems (ITS). Springer.

Systematics, C., 2005. Traffic congestion and reliability: Trends and advanced strategies for congestion mitigation. Technical Report. Federal Highway Administration, Texas Transportation Institute.

Tomasi, C., Manduchi, R., 1998. Bilateral filtering for gray and color images, in: Sixth International Conference on Computer Vision, IEEE. pp. 839–846. doi:10.1109/ICCV.1998.710815.

Yue, M., Fan, L., Shahabi, C., 2016. Inferring traffic incident start time with loop sensor data, in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, ACM. pp. 2481–2484. doi:10.1145/2983323.2983339.

Zhang, M., Gunturk, B.K., 2008. Multiresolution bilateral filtering for image denoising. IEEE Transactions on Image Processing 17, 2324–2333. doi:10.1109/TIP.2008.2006658.

Zhang, Y., Ren, J., Liu, J., Xu, C., Guo, H., Liu, Y., 2017. A survey on emerging computing paradigms for big data. Chinese Journal of Electronics 26, 1–12. doi:10.1049/cje.2016.11.016.

Zhang, Z., He, Q., Tong, H., Gou, J., Li, X., 2016. Spatial-temporal traffic flow pattern identification and anomaly detection with dictionary-based compression theory in a large-scale urban network. Transportation Research Part C: Emerging Technologies 71, 284–302. doi:10.1016/j.trc.2016.08.006.

Zhu, T., Wang, J., Lv, W., 2009. Outlier mining based automatic incident detection on urban arterial road, in: Proceedings of the 6th International Conference on Mobile Technology, Application & Systems, pp. 29:1–29:6. doi:10.1145/1710035.1710064.