AOID: Adaptive Online Incident Detection System

Arnold P. Boedihardjo, Chang-Tien Lu

Department of Computer Science, Virginia Polytechnic Institute and State University (Virginia Tech)

{aboediha,ctlu}@vt.edu

Abstract— **The provisions of any emergency management system with respect to the public safety necessitates the inclusion of the transportation network. The transportation network provides a means for mitigation strategies for any disaster, whether it is natural or human-induced. In this paper, we introduce a set of tools to integrate with a traffic information system to provide automatic traffic incident detection and traffic forecast. Current automated incident detection techniques may not perform well under changing traffic patterns, recurrent congestions, and may require large amounts of training data. We propose a solution to mitigate these shortcomings by utilizing predicted traffic models and performing comparative analysis against observed traffic patterns to automatically detect incidents.**

Index Terms— **automated incident detection, data mining and analysis, intelligent highway systems**

I. Introduction

The provision of any emergency management system with respect to the public safety necessitates the inclusion of the transportation road network. Obstructions to traffic flow (i.e., congestions) can enormously reduce a regions ability to transport inhabitants and resources. This hindrance debilitates the region from invoking optimal response strategies to emergent events. The transportation network provides a means for mitigation plans (e.g., evacuation) for any disaster, whether it is natural or human-induced. If the transportation network is compromised, then the safety of the public infrastructure can be fatally handicapped. Therefore, it is critical that a region provide supporting infrastructures to protect their transportation road network.

The effects of road congestions not only impact the region's infrastructure, but can severely hamper its economy and environment. It is estimated in 2005 that both recurrent and non-recurrent congestions cost $75 billion and dissipated approximately 8.4 billion gallons of wasted fuel [1]. The Texas Transportation Institute performed a nationwide study of 85 urban areas and reported that on average, an urban region (defined as a medium sized area) will lose $418 million due to congestion associated costs and produce 15 million gallons of wasted fuel [2]. The major contributors of congestions (more than half) are non-recurrent congestions. Non-recurrent congestions are results of incidents such as vehicular collisions, construction and maintenance activities, inclement weather conditions, or other activities which reduce the roadway capacity.

Several Intelligent Transportation Systems (ITS) have been deployed that employ mechanisms to minimize the damages ensued by roadway incidents. An important step in containing and reducing the damages caused by incidents is to minimize the time needed to respond to the event. ITS can employ a monitoring tool to observe the behavior of traffic within a region and provide an Automatic Incident Detection (AID) scheme to alert traffic operation personnel of a possible incident event. AID schemes are pivotal in providing maximum effectiveness for emergency personnel to react to an incident as it reduces the time to detect the event. For the victims of incidents, their chance of fatality rises 6% for every minute of delay it incurs on the emergency team's response time [5]. Furthermore, the chance of secondary incidents rises as the duration of an incident increases and travel times for commuters suffer due to incidents. A tool that ITS can leverage for performing road safety analysis is traffic trend prediction, which provide traffic planners and other key decision-makers insights into building effective roadway designs. These decisions can impact the ability of a roadway to handle sudden increased loads and moderate traffic capacity to reduce recurrent and non-recurrent congestions.

Much research and development have been emphasized on the subject of traffic incident detection. But many current methods have drawbacks that can render them ineffective for use in emergency management systems. Due to computational time constraints and their inability to automatically adapt to changing traffic behaviors, the current set of AID may not be suitable for use in the unpredictable and highly dynamic setting of emergency management systems.

Over the past two decades, several automatic incident detection methodologies have been proposed, such as [8], [9], [10], and [12]. These methods are able to determine incidents with high detection rates, but must be performed within strict environmental parameters such as specific temporal constraints and minimum traffic volumes. When the road behavior changes, approaches such as these may become less effective and require manual re-adjustments to their parameters. Other methods that can work under changing road conditions are based on computational intelligence. Many of these schemes utilize neural networks, Bayesian networks, and fuzzy logic [3, 7, 11]. But, these techniques can require large training datasets (which may not be readily available) and in some cases incur high runtime cost that cannot meet the computational constraints of emergency management systems.

The AID we have developed in this paper can adapt to the varying dynamics of the environment, requires a minimal set of training data which is a user-adjustable parameter, and efficiently utilizes the system's computational resources.

Adaptive ***O***nline ***I***ncident ***D***etection (AOID) is our proposed contribution to alleviate these deficiencies of current AID approaches. The AOID system is implemented as an extension to our ***A***dvanced ***I***nteractive ***T***raffic ***V***isualization ***S***ystem (AITVS) [6]. The AITVS, developed by Virginia Tech's Spatial Data Management lab, is a comprehensive traffic visualization system that presents summarizations of spatiotemporal patterns of road detector data in the Metropolitan Washington D.C. areas. AITVS marries a wide set of the multidimensional visual components with efficient processing algorithms to deliver a responsive and complete traffic visualization system. The AOID contributes and integrates two sets of tools into AITVS: recurrent traffic behavior forecasting and automatic incident detection. These tools supplement AITVS to give emergency personnel the information required to quickly devise and invoke an emergency plan. Furthermore, the information derived from the tools can be interfaced with traveler information systems such as highway variable signs to give commuters the most up-to-date traffic conditions.

The paper is organized as follows. Section II gives a detailed description of AOID. Section III explains our implementation and case study. And section IV provides our conclusion and future work.

II. Proposed Approach: AOID

AOID is our unified solution that targets the inter-dependent requirements of incident detection and traffic forecast. Traffic incident can be qualitatively described as follows: a *traffic incident* exhibits as a spatial anomaly that diverges for some threshold *m* (e.g., standard deviation for our case) from the *forecasted traffic* value. Since only a subset of the spatial anomalies is incident related, it is necessary to provide a classification mechanism to decipher which of the observed anomalies are representatives of actual incidents. A traffic trend model is generated in real-time and can adapt to evolving recurrent traffic patterns that are induced by changes in environmental parameters (e.g., long-term road construction, road expansions). Using the traffic trend models, the system determines the deviations of currently observed traffic values. The deviations are analyzed using statistical methods and processed into a dynamic decision-tree to classify incident occurrences. Hence, the architecture is divided into the following subtasks: **data preprocessing**, **model formulation**, **anomaly detection**, **classification**, and **data visualization**.

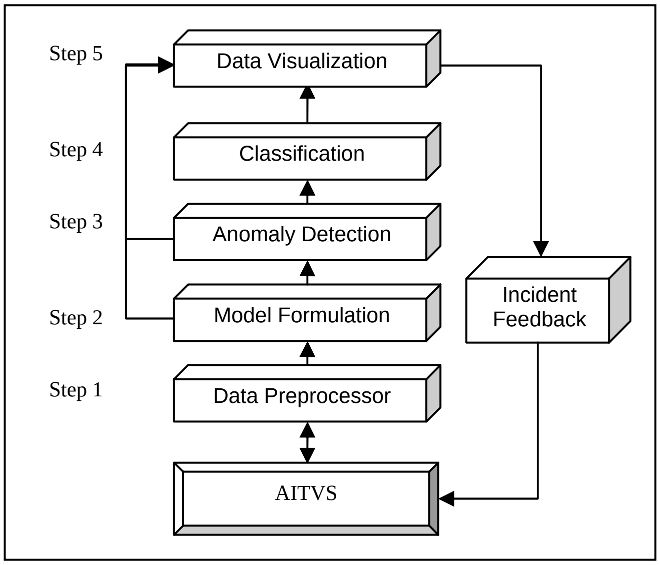

Figure 1: Architecture of AOID.

In the AOID (see Figure 1), detection is performed as follows. Firstly, a traffic trend model is generated for the given day. Then that day's observed traffic value is compared to the generated model and determined if anomalies exist within that observation. If anomalies are found, then they are classified with their incident probabilities. The AOID system is tightly coupled with the AITVS and as such it inherits all the low-level data preprocessing and access functionalities of the AITVS. Raw sensor data arriving at the AITVS server are parsed and stored in the AITVS traffic database. AOID utilizes this database, but the data are not filtered for noises such as sensor fluctuations and missing values. Therefore, it is the task of the **data preprocessor** in the AOID to furnish usable and cleaned traffic data to the remaining AOID subsystems. From the data preprocessor, traffic data is passed to the **model formulation** module to generate the traffic models in real-time. Then current traffic observations are sent to the **anomaly detection** module along with the generated traffic models. If anomalies are found, then this anomaly dataset is sent to the **classification** module. The classification gives the incident probability of the anomaly set.

Data Preprocessor (Step 1)

To perform the mining tasks, historical traffic data will need to be cleaned for noise, marked for incident related data, and labeled for short-term traffic events. Data noises include malfunctioning sensors and missing values. Marked or labeled records will be used to differentiate data stemming from normal traffic activity. The data preprocessing task is outlined as follows:

- **Noise filtration:** Scan the database for non-conforming records that are likely to originate from malfunctioning detectors or invalid values and remove these records
- **Incident filtration:** Find all records that are known to be associated with incident events and mark this information. These events include vehicular collisions, highway maintenance, and other incidents reported by the DOT database.
- **Short-term event filtration:** Find all records that are associated with holidays or special events and mark this information
- **Output:** Furnish traffic records that are associated with normal and recurrent traffic behavior

A rule-based algorithm is used to identify and remove data noises. Incident related data and short-term events will need to be correlated with the incident report and traffic event database which are performed using database join routines.

Join routines are costly functions that can take O(*n**log *m*) time complexity where *n* and *m* are the dataset cardinalities. Hence, to improve the performance of the preprocessing step, we propose an incremental approach for AOID. The AOID utilizes an agent that monitors newly arriving data in AITVS. The agent processes the new data at designated intervals to perform the necessary filtration tasks. The processed data are then re-inserted into the AITVS database for use with the model formulation and anomaly detection tasks of AOID. Therefore, preprocessing only occurs with newly arrived data.

Model Formulation (Step 2)

This module generates weekly traffic trends (e.g., traffic models) for each day of the week. For each day, the model captures the same set of days of some preceding weeks to generate the predicted trend. For example, to generate the traffic model for Monday of the 4th week of July, the module will obtain the traffic data from Monday of the 3rd, 2nd, and so forth of July and possibly extending to June. The number of weekly data is a user-adjustable parameter. After obtaining the past weekly data, the model will use a weighted average to produce the expected value (i.e., trend value). Weighted average will allow the model to adapt to changes in traffic patterns. The generated models will be used to determine deviating values of current traffic. Depending on the scope of the analysis, the models can be directly sent to the to the **data visualization** module for further studies.

To describe the traffic trend generation, we first define the following:

- Let $F(s,w,d,t)$ be the function that evaluates the value on week, w, day, d, time-step, t, and station, s.
- Let $V[i..n] = [F(s,w_1,d,t), F(s,w_2,d,t) \ldots F(s,w_i,d,t) \ldots F(s,w_n,d,t]$ where w_i is the week prior to w_{i-1} and n is the number of weeks
- Let α_i be the scalar weights given to each element of V

Then the predicted value is defined as follows $F(s,w_{predicted},d,t) = \alpha_1 V[1]+ \alpha_2 V[2] + \ldots + \alpha_n V[n]$.

Constraints for α_i :

- Give higher weights to more recent data i.e., $\alpha_1 > \alpha_2 > \ldots > \alpha_l \ldots > \alpha_{\mathbf{n}}$
- Make inter-weight relationship adjustable via a parameter (i.e., learning rate), θ
- Summation property:

$$\sum_{i=1}^{n} \alpha_i = 1 \tag{1}$$

Because the model needs to adapt to changes in traffic pattern behavior, it will necessarily assign higher weights to more recent data. In general, we can choose any formula for α_i that satisfies the above requirements. Let α'_i be weights that follow all of the given requirements above except for the summation property and the summation of α'_i is greater than 0. Then we make the following observation for α'_i:

$$\alpha_i = \left[\sum_{i=1}^{n} \alpha'_i\right]^{-1} \alpha'_i \tag{2}$$

Equation (2) relaxes the constraints of α_i and simplifies our search for formulae that will satisfy our requirements. For this paper, we have chosen two formulas for α'_i. One based on Zipf's law [4] and the second based on linearly proportional weights with respect to data recency. Zipf's law is defined as:

$$\sum_{i=1}^{n} \frac{n}{i^{\theta} H_{\beta}(\theta)} = n \tag{3}$$

$$H_{\beta}(\theta) = \sum_{j=1}^{\beta} \frac{1}{j^{\theta}} \tag{4}$$

For equations (3) and (4), n is the magnitude of V and H_β is the harmonic number with order θ of β. By assigning α'_i to (3), we have the following:

$$\alpha'_i = \frac{n}{i^{\theta} H_{\beta}(\theta)} \tag{5}$$

Derivation of α_i is achieved by using (2). Therefore, Zipf's law will give us the following α_i:

$$\alpha_i = \frac{1}{i^{\theta} H_{\beta}(\theta)} \tag{6}$$

Equation (6) does not give us a gradual and linearly proportional learning rate due to the definition of Zipf's law. Instead, it will give us an approximated method to change the weights' curve behavior. For example, by changing the θ parameter we can set each α_i with exponential distribution, inverse linear distribution, etc.

For our second formula, we will define α_i so that the weights are gradual and linearly proportional (to data recency) relative to their neighboring weights. To derive this formula, let us define a parameter, k, such that k is a scalar multiple of the oldest weight and that k will be the value assigned to the most recent weight. Therefore, the relationship of α'_i and its older neighbor can be described as follows:

$$\alpha'_i - \alpha'_{i+1} = \frac{k}{n-1} \tag{7}$$

Removing the recurrence relationship of equation (7) to a direct expression formula we have the following:

$$\alpha'_i = \alpha'_n + (n-i)\frac{k}{n-1} \tag{8}$$

Let $\alpha'_n = 1$ and we determine the closed-form summation formula for α'_i as follows:

$$\sum_{i=1}^{n} \alpha'_i = \frac{n(kn + n - k - 1)}{2n - 2} \tag{9}$$

Using equations (2), (8), and (9), we have the following final closed form solution for the linear weights α_i:

$$\alpha_i = \left(\frac{2n-2}{n(kn+n-k-1)}\right)\left(1 + (n-1)\frac{k-1}{n-1}\right) \tag{10}$$

If we let $k = \theta$ then we have established k as the learning rate parameter for formula (10). Both formulae (10) and (6) have been implemented as weights for the AOID.

Anomaly Detection (Step 3)

The task of this module is to determine anomalies for the currently observed values. Along with the traffic trend model, the variance of the traffic model is also provided by the **model formulation** step. This variance will provide the threshold deviation value at each time slot for classification of observed traffic as normal. Else, if the new value at that time slot deviates more than associated variance (threshold) then this new value is considered an anomaly. The variance is determined by calculating the standard deviation which is calculated as follows:

$$SD = (1/n) * SQRT\ (\alpha_1\ (V[1]-F(s,w,d,t+1))^2 + \alpha_2\ (V[2]-F(s,w,d,t+1))^2 + \ldots + \alpha_n\ (V[n]- F(s,w,d,t+1))^2\ ) \quad (11)$$

The deviation between current observation and trend value, *D*, is defined as follows:

$$D = |F(s,w_{predicted},d,t)\ -F(s,w_0,d,t)| \quad (12)$$

Therefore, the current observation, ***F(s,w***$_0$***,d,t)***, is an outlier if its *D > SD*.

This task highlights the importance of the preprocessing stage. Because the calculation of *SD* and *D* can be highly sensitive to perturbations in data behavior, it is important that the data used in the model generation are free from noise and other knonwn anomalies (e.g., incidents). It is also equally important to consider and remove noises from *current* traffic data otherwise *D* can include anomalies that are defined in the preprocessing step and may be misclassified in the latter steps.

Incident Classification (Step 4)

This module verifies that the anomalous data received from step 3 reflect actual incidents. The anomalies are aggregated and compared to known incidents. Then a similarity score is assigned and designated as incident probabilities. The similarity score is calculated in two phases:

- Anomalous events are assigned weights with respect to their closeness to neighboring anomalies (with respect to time)
- Incident probability of the anomalous dataset is calculated based on its weight and distance to its incident vector

Let *W* represent the weight for an anomaly set *S*, then *W* is increased by some positive valued function, *f*, if and only subsequent anomalous data is close in time with the last occurring element in *S*. Therefore, larger clusters of anomalous data will receive higher *W*. The clusters and along with their weights are further analyzed to determine the similarity score with an *incident vector*. An incident vector contains descriptive components to describe a particular class of incidents. Primarily, the components are the mean deviation in occupancy, speed, and volume. This approach gives AOID the ability to classify incidents into specific types. For example, the incident vector for inclement weather will be markedly different from the incident vector associated with vehicular collisions. The similarity measure (i.e., incident probability) used in our approach is simply one minus the ratio of the distance of current traffic vector from the incident vector and the distance of trend vector from the incident vector. We give the incident probability measure below:

Incident Probability = 1 - Distance(Observed,Incident) / Distance (Trend,Incident) (13)

We employ the Euclidean distance as our distance measure. Euclidean distance is a simple and elegant approach for several similarity rankings, but can be problematic if some subsets of the data dimensions are multiply correlated on varying degrees. Hence, we introduce a scaling list for each of the incident vector. For example, components that are highly and positively correlated will be assigned with smaller scales since they do not belong to the set of components (i.e., principal components) that define a particular incident class. The scaling list can be regarded as an approximate variance-covariance matrix in the Mahalanobis distance. For our current dataset, the Euclidean distance with its scaling list is suitable for our purposes since an incident vector contains only three components. In addition, determining the Euclidean distance with its scaling list incurs a small cost that is proportional to the number of components in a vector. However, the Mahalanobis distance requires that the inverse variance-covariance matrix be computed which can incur a cost that is quadratic to the number of components. But as the size of the incident vector becomes large (e.g., more data types), the scaling list will no longer be suitable to represent the variability and correlation of the vector components. At which point we will need to consider adopting the Mahalanobis distance for AOID and investigate approaches that optimize its computation.

Data Visualization (Step 5)

This module serves as an interface to the user and various subsystems of AOID. The module creates a graphical representation from the following components: model formulation, anomaly detection, and classification. For the model formulation, volume, speed, and occupancy plots for both the currently observed traffic patterns and the generated traffic models are shown (Figure 2a). Lastly, incident probabilities of the currently observed values will be plotted on a time series graph (Figure 2b).

Parameter Tuning

The learning rate parameter (i.e., θ), the *vector scales* in the incident classification step, and the incident probability threshold (i.e., alarm value) are determined by applying heuristics from the training dataset. The learning rates are calculated by selecting a maximum historical data range, processing those historical samples from the training set, and calculating the θ values which optimizes model accuracy. For each incident class and their instances, the correlation values are calculated to give the class' *vector scales*. Similarly, the incident probability threshold is determined for each incident class by taking the average probability values exhibited by each instances of the class.

III. Implementation and Case Study

The AOID is implemented using Java 1.4.2 and designed as a subsystem of AITVS. AITVS provides interfaces for traffic data access and visualization. Low-level data processing is performed by the AITVS. Tasks such as data fusion and translation are dispatched within the AITVS to a standard format that can be efficiently accessed by the AITVS subsystems. For our case study we use incident cases of I-66 from January 2004 to December 2005.

Real-time application: Adapting our approach for real-time environment follows a stream data mining technique in which AOID will incrementally determine the outlier values and hence their incident classification for the newly arriving data by using the synopsis information given from prior weeks. Because **anomaly detection**, **incident classification**, and **incident feedback** can be efficiently done in real-time, these components will not require any algorithmic or design alterations for application in a real-time scenario. However, the **model formulation** will need to be re-adapted to serve as a summarization (synopsis) structure for which to compare current arriving data. Since the model formulation only requires information from *past records* in the database, it can be generated before the real-time monitoring task is invoked. For example, real-time monitoring for i^{th} day will utilize the summarization structure that has been pre-generated at some previous j^{th} day (i.e., $j < i$).

In the following we give in-depth discussions of three cases, one in 5/3/2005 at station 331 (case 1), second in 7/24/2005 at station 341 (case 2), and third in 5/1/2005 at station 261 (case 3). For case 1 (Figure 2), we observe that an incident occurred between 10:15AM to 10:30PM with a peak probability of approximately 0.95. We verify this fact using the Virginia Department of Transportation (VDOT) incident database. For this case, VDOT reported that a vehicular collision occurred at 10:16AM and cleared at 10:27AM which coincides with the first spike in the incident graph. This spike is indicative of the initial impact of a vehicular collision. At initial impact, immediate trailing vehicles will reduce their speeds dramatically and form an anomalous cluster dataset which is translated to the first spike of the incident probability. But oncoming traffic will slow down (regardless of the fact that the incident has been cleared) due to debris or other vehicles slowing down which accounts for the presence of a second spike. This second spike is not directly indicative of a separate incident but rather a subsequent congestion (i.e., secondary incident) that resulted from the first incident. Because this is a non-recurrent congestion, the AOID will regard this as an incident and could potentially cause a false alarm. One way to remedy this issue is to adjust the weights of the anomalous dataset such that subsequent anomalous clusters are assigned with lower weights.

In case 2 (Figure 3), we demonstrate the effectiveness of the AOID for identifying incidents in the presence of non-incident anomalous data. Between 3:30PM and 6:30PM there is divergence in traffic behavior as occupancy is higher than expected. The AOID views this as an anomalous set but is dismissed at the classification stage as it does not exhibit the behaviors of an incident. But, at approximately 10:30PM during low volumes, the AOID shows that there is a 0.85 probability that an incident occurred.

In case 3 (Figure 4), it is observed that an incident occurred at around 1:30PM-7:00PM. Similar to the above cases, the duration of the incident for case 3 can also be evaluated by estimating the length of the incident probability graph for those values that exceed an incident probability threshold. The gap between 2:30PM-3:00PM is indicative of the situation that is explained in case 1. Prior to the incident, at approximately 10:00AM to 11:45AM there is a dramatic drop on the volume curve due to a malfunctioning detector. Much of this data is detected and filtered at the preprocessing stage. However, its neighboring data anomalies (Figure 4a) at 9:30AM-10:00AM and 11:45PM-12:00PM, were not removed but transferred onto the remaining AOID components. At the classification stage, these anomalous datasets are assigned low similarity scores due to their large distances from the incident vectors, and hence become categorized as non-incidents. This demonstrates a critical step in removing all such potential false alarms.

We used 8 actual incident cases to evaluate the AOID and in each case the AOID is able to detect the incidents with probabilities higher than 0.70. Because of the limited number of incident cases, detection rate and false alarm rate metrics were not considered. However, the paper provides a set of case studies covering a wide scope of incident types to observe and validate AOID's approach.

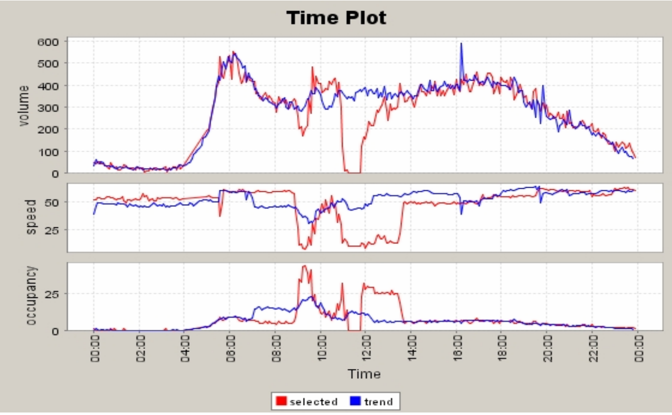

Figure 2(a): Traffic for (5/3/2005, EB, 331).

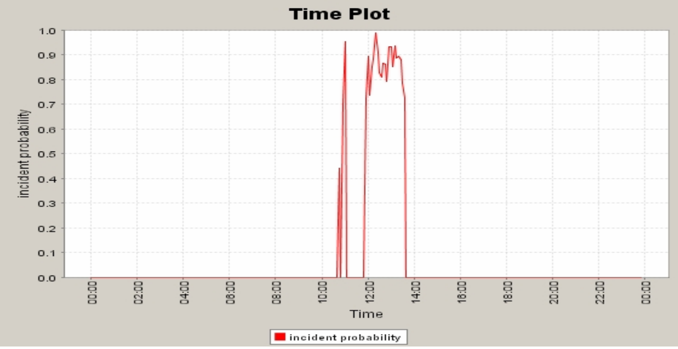

Figure 2(b): Incident results for (5/3/2005, EB, 331).

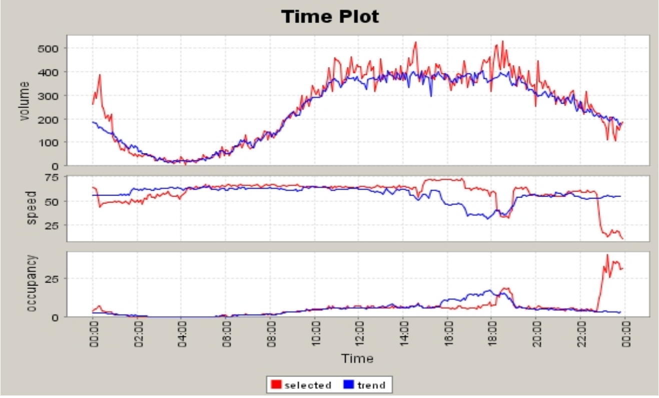

Figure 3(a): Traffic for (7/24/2005, EB, 341).

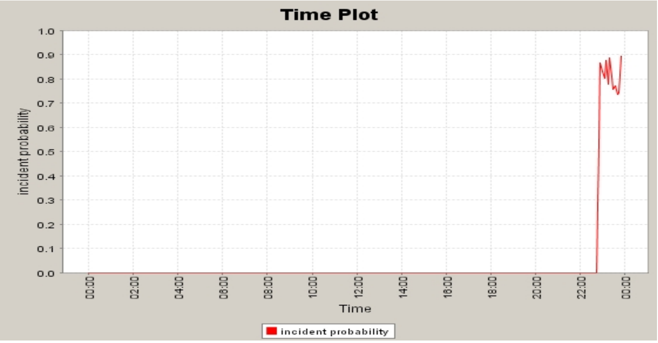

Figure 3(b): Incident results for (7/24/2005, EB, 341).

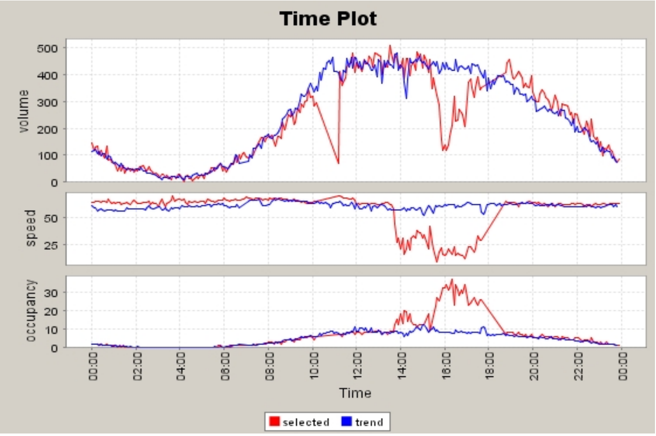

Figure 4(a): Traffic for (5/1/2005, EB, 261).

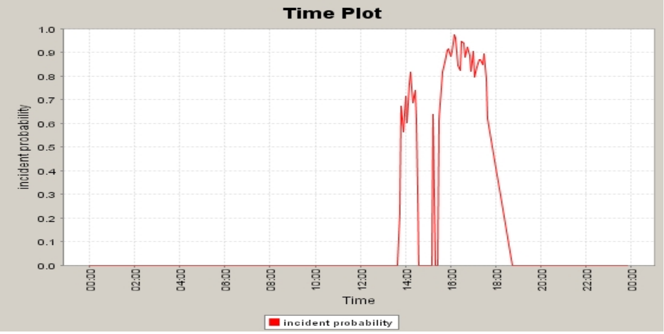

Figure 4(b): Incident results for (5/1/2005, EB, 261).

IV. Conclusion and Future Work

In this paper, we introduced two sets of traffic analysis and monitoring tools: traffic trend prediction and automatic incident detection. The tools are integrated to form the AOID system which detects incidents within various and dynamic traffic environments, requires minimal training data set, and efficiently uses the system's computational resources. We also developed an effective learning algorithm using Zipf's law and a linearly proportional weight scheme to generate the traffic models and predictions.

Future work includes further experimentation to determine the optimal range of anomalous dataset weights and refine the incident classification technique to cover a wider range of incident types. Incidents may occur between stations and more accurate detection can be derived by fusing nearby stations' data, therefore another future work of this project is to investigate efficient methods that integrate the stations' spatial properties.

V. References

[1] "Incident Management: Detection, Verification, and Traffic Management," U.S. Department of Transportation Federal Highway Administration, *http://www.fhwa.dot.gov/tfhrc/safety/pubs/its/pabroch/fotincident.pdf*, 1998.

[2] "What does congestion cost us?," Texas Transportation Institute, *http://mobility.tamu.edu/ums/report/congestion_cost.pdf,* 2003.

[3] H. Adeli and A. Karim, "Fuzzy-wavelet RBFNN model for freeway incident detection," *Journal of Transportation Engineering*, vol. 126, pp. 476-471, 2000.

[4] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*: ACM Press, pp. 146-147, 1999.

[5] M. Evanco, "The Impact of Rapid incident Detection on Freeway Accident Fatalities," Mitretek Systems Inc., 1996.

[6] C. T. Lu, A. P. Boedihardjo, and J. Zheng, "AITVS: Advanced Interactive Traffic Visualization System," *To Appear in Proceedings of The 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, 2006.

[7] I. Ohe, H. Kawashima, M. Kojima, and Y. Kaneko, "A method for automatic detection of traffic incidents using neural networks," *In Proceedings of The Vehicle Navigation and Information Systems Conference*, pp. 231-235, 1995.

[8] H. J. Payne and S. C. Tignor, "Freeway incident detection algorithms based on decision tree with states," *Transportation Research Record*, vol. 682, pp. 378-382, 1978.

[9] B. N. Persaud, F. L. Hall, and L. M. Hall, "Congestion identification aspects of the McMaster incident detection algorithm," *Transportation Research Record*, vol. 1287, pp. 167-175, 1990.

[10] O. Sawaya, "Real-time Incident Traffic Management Methodologies," in *Civil Engineering*. Evanston, Illinois: Northwestern University, 2000.

[11] D. Srinivasan, R. L. Cheu, and Y. P. Poh, "Hybrid Fuzzy Logic-Genetic Algorithm for Automated Detection of Traffic Incidents on Freeways," *In Proceedings of The IEEE Intelligent Transportation Systems Conference Proceedings*, pp. 352-357, 2001.

[12] H. Xu, C. M. Kwan, L. Haynes, and J. D. Pryor, "Real-Time Adaptive On-Line Traffic Incident Detection," *In Proceedings of The IEEE Symposium on Intelligent Control*, pp. 200-205 1996.