# Traffic Incident Validation and Correlation Using Text Alerts and Images

Wye Huong Yan, Justin Ong, Shen-Shyang Ho
School of Computer Engineering
Nanyang Technological University
Singapore
{whyan1,jong2}@e.ntu.edu.sg, ssho@ntu.edu.sg

Jim Cherian
BMW-NTU Future Mobility Research Joint Laboratory
Nanyang Technological University
Singapore
jcherian@ntu.edu.sg

## ABSTRACT

One of the major challenges during the process of extracting information from multiple spatio-temporal data sources of diverse data types is the matching and fusion of extracted knowledge (e.g. interesting nearby events detected from text, estimated density or flow from a set of geo-coded images). In this demonstration, we present PETRINA ("PErsonalized TRaffic INformation Analytics"), a system that provides traffic-related incident monitoring, mapping, and analytics services. In particular, we showcase two main functionalities: (1) text traffic alert validation based on traffic condition information derived from traffic camera images and (2) traffic incident correlation based on spatio-temporal proximity of different incident types (e.g., accidents and heavy traffic). Despite the fact that the images are sparse (available every three minutes), the regularity makes it possible to validate whether a text traffic alert is outdated or not, and to more accurately estimate the time elapsed and total incident time. Multiple traffic incidents can be grouped together as a single event based on the traffic incident correlation to reduce information redundancy. Such enhanced real-time traffic information enables PETRINA to offer services such as dynamic routing with traffic incident advices, spatiotemporal traffic incident visual analytics, and congestion analysis.

## Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services; J.7 [**Computers in Other Systems**]: Real time

## General Terms

Design

## Keywords

Traffic data, Data Analytics, Spatiotemporal data mining

## 1. INTRODUCTION

With the ubiquitous availability of online mapping services, one can search for directions from point A to B easily. However, roads are not always free from traffic incidents such as road works, accidents, heavy traffic, and vehicle breakdowns. Although several online mapping services do provide traffic incident alerts, they may often display the entire set of ongoing traffic incidents regardless of the traffic incidents that are relevant to the chosen route [1] [2]. In many cases, the traffic incident alerts are not updated on time since the update process requires manual validation of incidents (e.g. traffic police to report a new incident or that normalcy has been restored). As a result, two special scenarios can occur: (1) traffic incident has expired, but traffic alert still exists, and (2) traffic incident is still active, but there is no traffic alert. Moreover, some incidents can be mutually dependent and could be reduced to a single event. For example, heavy traffic caused by an accident or a vehicle breakdown, which can be represented through a merged event.

In this paper, we handle the above scenarios as a matching problem on the extracted knowledge from the text-based traffic alerts and traffic camera images available from the Singapore Land Transport Authority (LTA). We propose two functionalities: (i) text traffic alert validation based on traffic condition information derived from traffic camera images and (ii) traffic event correlation based on spatiotemporal proximity of different event types. The two functionalities can work on both real-time and historical data. The main challenges are, namely: (a) accurate traffic flow estimation from sparse camera images and (b) efficient event-event and event-image matching. The two capabilities are integrated into PETRINA ("PErsonalized TRaffic INformation Analytics"), a prototype system that provides traffic-related incident monitoring, mapping, and analytics services. In spite of the sparsity of images, the regularity of available images makes it possible to validate whether a text traffic alert is outdated and to estimate traffic incident time elapsed and total incident time more accurately. Multiple traffic incidents can be grouped together as a single event based on the traffic incident correlation to reduce information redundancy. These enhanced traffic event information enable PETRINA functionalities such as routing with traffic incident advice, spatiotemporal traffic incident visual analytics, and congestion analysis.

The rest of this paper is organized as follows. In Section 2, we provide a brief overview of previous work. In Section 3, we provide


SIGSPATIAL '14, Nov 04-07 2014, Dallas/Fort Worth, TX, USA
ACM 978-1-4503-3131-9/14/11.
http://dx.doi.org/10.1145/2666310.2666379

[1] One Motoring, Singapore's premier motoring site for LTA eServices, motoring and traffic information. http://interactivemap.onemotoring.com.sg/mapapp/.

[2] One Map, an integrated map system for government agencies to deliver location-based services and information. http://www.onemap.sg/index.html.

an overview of the PETRINA system components. In particular, we describe how raw text and image data are processed. In Section 4, we describe the matching process in detail and then illustrate the image-based text traffic alert validation and the traffic event correlation capabilities using some examples.

# 2. RELATED WORK

There have been many works and systems that try to address the problem of automatic detection of traffic incidents on highways through vision-based methods. Spatial correlation of text and imagery has also been researched, for example by relating the geographic locations of the events referenced in text with images acquired at the same location [6]. However, our focus has been on considering the spatio-temporal correlation of traffic incident-specific text data with corresponding images. To estimate traffic speed and condition, unlike previous work [2] (and reference therein) using video sequences, our methods are based on images that are sparse and of low sampling frequency.

# 3. PETRINA

The current prototype PETRINA focuses on Singapore traffic data and is designed as a web application that can be run on a desktop or laptop computer. PETRINA retrieves near-live traffic data streams in XML format from the Singapore Land Transport Authority (LTA)[3]. The collected Text and image data are processed to extract important information such as image binary data, keywords, toponyms, timestamps, and geo-location. Subsequently, the data is structured and stored into a database. The database accumulates traffic information to perform analytics on both historical data and near real-time data. PETRINA consists of the following main components: (i) Text Traffic Alert Stream, (ii) Traffic Image Stream, (iii) Spatiotemporal Traffic Event Database, (iv) Analytic Function Library, and (v) Web Server shown in Figure 1. The database and web server implementation are not described here due to page limitation. The Analytic Function Library consists of analytic functions and applications discussed in this paper[4]. Next, we describe the first two components in detail.

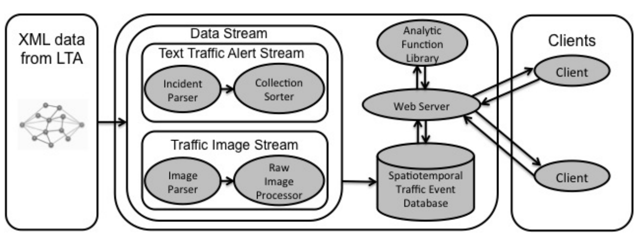


**Figure 1: PETRINA System Components.**

## 3.1 Text Traffic Alert Stream

The Text Traffic Alert Stream is responsible for establishing HTTP connection to LTA's data mall, raw text traffic alert data retrieval, and text processing. Raw XML traffic data undergoes a multi-step process of information extraction. Each XML file contains several sub-trees, one per traffic incident alert. Example 1 and 2 shows traffic alerts for two different traffic incident types. The message include (1) the date and time when the incident occurs, and (2) the location and traffic direction where it occurs.

**Table 1: Incident Hash Map**

| Key | Value |
|---|---|
| on | PIE |
| towards | Tuas |
| after | Jurong East Ave 2 |
| avoid | lane 3 |

EXAMPLE 1. *Information from a vehicle breakdown alert.*

- Message: *(22/5) 20:01 Vehicle breakdown on PIE (towards Tuas) after Jurong East Ave 2. Avoid lane 3.*
- Latitude: *1.4368240216197663*
- Longitude: *103.76840500188547*
- Type: *Vehicle Breakdown*

EXAMPLE 2. *Information from a heavy traffic alert.*

- Message: *(17/7) 7:40 Heavy Traffic on TPE (towards SLE) between Punggol Rd Exit and CTE Exit.*
- Latitude: *1.4006892196293519*
- Longitude: *103.85913887287498*
- Type: *Heavy Traffic*

There are four traffic incident types, namely: accident, vehicle breakdown, roadwork, and heavy traffic. However, for a heavy traffic incident (see Example 2), only a single point location is available although the message describes a long stretch of road. Moreover, there is no information about when a traffic incident is cleared, and also no information on whether two traffic incidents are correlated. We will analyze these issues in Section 4.

The Text Traffic Alert Stream module is made up of two components: (i) Incident Parser and (ii) Collection Sorter.

### 3.1.1 *Incident Parser*

A hash map is used to store spatial contextual information extracted from the message word array (see Example 1). A keyword library which contains prepositions and traffic-related words or phrases are used to match the non-toponymic words in the message. It contains words such as "at", "on", "towards", "in", "after", "before", "near", "between", and "congestion". When a keyword is found in a message, the index of the matched keyword is stored into a keyword-index array in the form of $\{keyword, i\}$ where $i$ is the index. After the keyword matching process, the array is sorted in an index ascending order. Then, we perform toponym recognition and resolution [1] for word index (or phrase indices) after a keyword index or between two keyword indices. For instance, the message in Example 1 would have keywords "on", "towards", "after" and "avoid" that match the word array so that the keyword-index array is {{on, 3}, {towards, 5}, {after, 7}, {avoid, 12}}. The recognized toponyms are stored into the *Incident* object under a hash map utilizing keywords as the unique keys assuming that a keyword cannot appear more than once in a message (see Table 1). One notes that a traffic incident is unlikely to spatially coincide with the recognized toponym in the message. For example, the latitude and longitude values in Example 1 will not match those of the recognized toponym "Jurong East Ave 2". Geocoding can convert an address

[3]LTA data mall.
http://www.mytransport.sg/content/mytransport/home/dataMall.html
[4]4-minute video demonstrating the functionalities in PETRINA is available on http://vimeo.com/73961230

to a latitude and longitude pair, but the derived geographic position coordinates may not be very accurate. Since the precise street address is not provided, the derived position will be coarse.

An *Incident* object also stores attributes such as latitude, longitude, incident type, date and time of occurrence. Incident time elapsed is a derived attribute computed by monitoring and matching the same events over a period of time, handled by the collection sorter that we discuss next.

#### 3.1.2 Collection Sorter

When the collection sorter receives traffic incident data objects, these objects are checked to determine which collection they belong to, either ongoing or historical. The former collection stores all current incidents while the latter stores all the expired incidents. Data retrieval is made efficient by splitting traffic incidents into two separate collections instead of distinguishing them by a boolean property. When new incidents are received, their ID values are compared against existing entries and only new incidents get added into the ongoing collection. Incidents that no longer occur are removed.

One notes that since such text-based traffic alerts get updated and released through sporadic manual validation, the expired incidents may continue to be published for a long duration. Hence, even if an alert is highly accurate when first announced, a user cannot rely on it's correctness or validity without obtaining supplementary information from other data sources.

### 3.2 Traffic Image Stream

Similar to the Text Traffic Alert Stream, the Traffic Image Stream connects using HTTP to LTA's data mall to retrieve XML data. A major difference is that the raw images are published without any human editing or validation. Unlike previous work [2] (and reference therein) that utilizes video sequences to estimate traffic speed and condition, the images from LTA are sparse and of low sampling rate (i.e., updates every three minutes).

#### 3.2.1 Image Parser

We utilize images from forty-nine stationary traffic cameras spread across Singapore. Each image has an ID, creation date, location (based on camera position), and a URL to a static image taken at the current time of the day. For each image, the Image Parser creates a RawImage object based on these attributes and stores the objects into the RawImage collection in the database. Figure 2 shows two traffic instances from two different cameras.


**Figure 2: Examples of Traffic Image.**

#### 3.2.2 Raw Image Processor

The main objective of this process is to classify the traffic condition that appears in a camera image into one of the three categories, namely heavy (congestion), medium (normal), and light. We use an edge-detection approach to identify vehicles on the road [3] and estimate the vehicular density on a particular stretch of road visible from the image based on the density of edge pixels. We perform the well-known morphological gradient approach [4] to detect edges in the original image based on the difference between its dilated and eroded images. To enable accurate vehicular density estimation, non-road regions (e.g., trees) and opposite direction of the targeted road (see Figure 2) are masked out using OpenCV subtraction with manually designed mask images. The edge pixel counts on the road region are then used to estimate and classify the traffic condition on the road.

For each camera location, we collect and manually annotate a total of ninety images such that thirty images are collected for each category. Experimental results based on three-fold cross-validation have shown that selecting threshold values at the thirty-third and sixty-sixth percentile of image edge pixel counts from the training set to group the traffic images into the three categories achieves an average classification accuracy of 80% on test images. Note that the threshold values vary for different locations. A major factor that causes mis-classification is the presence of tree shadows in images (see right image in Figure 2).

## 4. EVENT VALIDATION AND CORRELATION

We describe the matching process in detail and then illustrate the image-based text traffic alert validation and the traffic event correlation capabilities using some examples.

### 4.1 Events and Images Matching

A key step for the validation and correlation functions is the matching (1) between two different events and (2) between an event and an image. In general, given two events (or an event and an image), one needs to check whether they satisfy the spatial and temporal requirements defined by a user. Here, we determine a match when the distance between two events (calculated by Haversine formula) is less than the user-defined spatial threshold, and their timestamps are close enough within the user-defined temporal threshold. There could initially be multiple matches for a particular event. To select the best match, we use the cosine similarity measure based on their timestamp and geographic position. These attributes are normalized using z-score normalization, with the mean and standard deviation of the traffic incident type of interest as the input. The resulting correlation values range from -1 (exactly opposite), to 1 (exactly same), with 0 usually indicating independence, and in-between values indicating varying degrees of correlation. Also, we assume no matching of same traffic incident type. For event and images matching, an event can be matched to a sequence of images based on the derived time-elapsed (see Section 3.1.2).

Once the matching process is done, one could use the matched results for text traffic alert validation and traffic event correlation.

### 4.2 Image-based Text Traffic Alert Validation

A text traffic alert is validated against the estimated traffic condition (derived from the matched images) by evaluating how spatiotemporally close the traffic camera is to the traffic incident, constrained by the user-defined spatiotemporal thresholds. The top record in Figure 3 shows the occurrence of two traffic incidents (that may be correlated, see Section 4.3), an accident and a vehicle breakdown, resulting in heavy traffic as shown in the left image in Figure 2. We continue to monitor the matched images until the traffic condition returns to normalcy and claim that the traffic incident is resolved. Both incidents can then be moved into the historical collection (see 3.1.2). This is further illustrated using the bottom record in Figure 3 with occurrence of two traffic incidents, (again)

| | | | | | | |
|---|---|---|---|---|---|---|
| 37 | Mon Aug 05 2013 18:25:00 GMT+0800 (SGT) | (5/8)18:25 Accident on SLE (towards BKE) after TPE Entrance with congestion till TPE Exit. Avoid lane 4. | Mon Aug 05 2013 18:30:00 GMT+0800 (SGT) | (5/8)18:30 Vehicle breakdown on SLE (towards BKE) after TPE Entrance. Avoid lane 4. | 5 mins | High (image) |
| 242 | Thu Aug 22 2013 17:03:00 GMT+0800 (SGT) | (22/8)17:03 Accident on BKE (towards Woodlands) after SLE Exit. | Thu Aug 22 2013 16:20:00 GMT+0800 (SGT) | (22/8)16:20 Vehicle breakdown on BKE (towards PIE) before Woodlands Ave 3 Entrance. | 43 mins | High (image) |

**Figure 3: (Two snapshots from PETRINA) The top and bottom matched traffic alert records correspond to the left and right images, respectively, in Figure 2.**

**Accident VS Vehicle Breakdown**

| | | | | | | |
|---|---|---|---|---|---|---|
| 112 | Mon Aug 05 2013 12:29:00 GMT+0800 (SGT) | (5/8)12:29 Accident on AYE (towards ECP) before Clementi Ave 2 Exit. | Mon Aug 05 2013 11:53:00 GMT+0800 (SGT) | (5/8)11:53 Vehicle breakdown on AYE (towards ECP) after Clementi Ave 2 Exit. Avoid lane 1. | 36 mins | High (image) |
| 113 | Mon Aug 05 2013 08:12:00 GMT+0800 (SGT) | (5/8)8:12 Accident on Lornie Road (towards Braddell) after PIE with congestion till Farrer F/O. Avoid right lane. | Mon Aug 05 2013 08:27:00 GMT+0800 (SGT) | (5/8)8:27 Vehicle breakdown on PIE (towards Changi Airport) after Eng Neo Ave. | 15 mins | High (image) |
| 114 | Mon Aug 05 2013 08:30:00 GMT+0800 (SGT) | (5/8)8:30 Accident on PIE (towards Jurong) after Toh Tuck Ave. | Mon Aug 05 2013 08:30:00 GMT+0800 (SGT) | (5/8)8:30 Vehicle breakdown on PIE (towards Jurong) after Toh Tuck Ave. | 0 mins | High |

**Figure 4: (Snapshot from PETRINA) Examples of traffic incident matches between types "Accident" and "Vehicle Breakdown", happening within 2 kilometers and 60 minutes of each other. (image) indicate the availability of processed traffic images; "High" indicates high correlation. Second last column reports the time difference between the timestamps for both incidents.**

an accident and a vehicle breakdown, and the right image in Figure 2 where the traffic is back to normal (or light traffic). Elapsed time and total incident time calculated are significantly improved compared to the case if only the traffic text alert data was used.

Accurate estimation of traffic condition from the images from stationary cameras allows us to automatically validate the correctness of the text traffic alerts. This enables the dissemination of more credible traffic alerts to drivers. However, the limited deployment of traffic cameras can be a practical issue.

### 4.3 Traffic Event Correlation

At some occasions, two traffic incidents may be related to each other. For example, an accident (or a vehicle breakdown) resulting in heavy traffic may get published as independent traffic alerts. It is preferable to merge them into a single event based on their spatiotemporal proximity to reduce information redundancy. Moreover, understanding the relationships between incidents is helpful in finding out their cause and effect. The relationships can be further refined to a spatiotemporal context. For example, one may query the correlation of the accident and heavy traffic incidents at a location during a particular time of the day.

The matching process and similarity (or correlation) calculation can be done efficiently either at near real-time or in future at user query time. Figure 4 shows the query outputs from highly correlated accident and vehicle breakdown incidents occurring in Singapore islandwide from August 1 2013 00:00 to September 6 2013 21:20. Although not shown in the figure, the query also returns the following information: (a) the total number of accidents (341), (b) the total number of vehicle breakdowns (1006), (c) the number of spatiotemporally correlated matches (106), and (d) the proportion of matches with respect to accidents (31.1%).

## 5. CONCLUSIONS

We present PETRINA, a system that provides traffic-related incident monitoring, mapping, and analytics services that processes real time raw traffic data and image streams. We highlight two functionalities: (1) text traffic alert validation based on traffic condition information derived from traffic images, and (2) traffic event correlation based on spatiotemporal proximity of different event types. Due to page limitation, functionalities such as routing with traffic incident advice, spatiotemporal traffic incident visual analytic, and congestion analysis that utilize the outcomes from the validation and correlation capabilities, are not described in this paper. They will be presented in our prototype demonstration[5].

## Acknowledgment

This project is supported by the NTU-SUG and AcRF Grant RG-41/12.

## 6. REFERENCES

[1] Shen-Shyang Ho, Mike Lieberman, Pu Wang, and Hanan Samet, *Mining future spatiotemporal events and their sentiment from online news articles for location-aware recommendation system*, Proceedings of the First ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems, pp. 25-32. ACM, 2012.

[2] Seon Ho Kim, Junyuan Shi, Abdullah Alfarrarjeh, Daru Xu, Yuwei Tan, and Cyrus Shahabi, *Real-Time Traffic Video Analysis Using Intel Viewmont Coprocessor*, Databases in Networked Information Systems, pp. 150-160. Springer Berlin Heidelberg, 2013.

[3] D. M. Ha, J-M. Lee, and Y-D. Kim, *Neural-edge-based vehicle detection and traffic parameter extraction* Image and vision computing, vol. 22, no. 11, pp. 899-907, 2004.

[4] James Lee, R. Haralick, and Linda Shapiro, *Morphologic edge detection*, IEEE Journal of Robotics and Automation, vol. 3, no. 2, pp. 142-156, 1987.

[5] Quannan Li, Yu Zheng , Xing Xie , Yukun Chen , Wenyu Liu, and Wei-Ying Ma, *Mining user similarity based on location history*, Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems, 34, 2008.

[6] Edward Waltz, Richard A. Berthiaume, *Text and imagery spatial correlator*, Patent US 7117192 B2, 2006.

[5] see http://vimeo.com/73961230 for other functionalities of PETRINA