

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

# Cloud-Assisted Mobile Crowd Sensing for Route and Congestion Monitoring

ÖZGÜN YILMAZ<sup>1</sup>, LEVENT GÖRGÜ<sup>2</sup>, MICHAEL J. O'GRADY<sup>2</sup>, (Senior Member, IEEE), and GREGORY M. P. O'HARE<sup>3</sup>, (Member, IEEE)

<sup>1</sup>Department of Computer Engineering, Ege University, 35100 Bornova, İzmir, Turkey (e-mail: ozgun.yilmaz@ege.edu.tr)

<sup>2</sup>School of Computer Science, University College Dublin, Dublin 4, Ireland

<sup>3</sup>School of Computer Science and Statistics, Trinity College Dublin, Dublin 2, Ireland (e-mail: gregory.ohare@tcd.ie)

Corresponding author: Özgün Yılmaz (e-mail: ozgun.yilmaz@ege.edu.tr).

**ABSTRACT** Accurate and reliable real-time urban traffic management can benefit urban citizens' daily life by reducing stress, travel time and carbon footprint. The provision of reliable and accurate traffic information has however proven to be a major challenge in intelligent transportation systems. Citizens carrying smartphones can be exploited as an important provider of traffic information and the mobile crowd sensing paradigm can be used as a solution to this challenge. In this paper, an urban traffic monitoring system which exploits the power of participatory sensing and cloud messaging is proposed. Crowd intelligence that is used to estimate traffic congestion levels, arrival times, and average road speed is harvested from crowd sensed data. Traffic congestion control at route level is implemented with a route guidance system. Proactive warnings or recommendations to drivers in the vicinity of or on the route to reported events are provided. The drivers can also report short-term traffic events and physical road conditions for road monitoring. Real-world experiments are conducted with a prototype implementation and the results demonstrates both system feasibility and traffic estimation accuracy.

**INDEX TERMS** Crowd intelligence, intelligent transportation system, mobile crowd sensing, real-time traffic management, traffic congestion

## I. INTRODUCTION

**T**RAFFIC congestion is a major challenge for our society. The sustainable growth of our cities is hampered by waste of energy, pollution and delays caused by traffic congestion in modern transportation systems [1], [2]. Intelligent Transportation Systems (ITS) seeks to make transportation more efficient and to provide solutions to the increasingly serious transportation problems [3], [4]. ITS aims at improving mobility, safety, efficiency, air quality, and decreasing energy consumption by providing real-time traffic information to drivers and traffic authorities [5]. Traffic prediction is a crucial aspect of an intelligent transportation system [6].

In conventional ITS, the traffic information is usually collected using specialized hardware and fixed sensors integrated within road infrastructure, such as loop detectors and surveillance cameras [7], [8], but these provide limited information [9]–[13]. Dynamic traffic information is typically displayed over message boards or

broadcast over the radio. Deployment and maintenance costs associated with such sensing infrastructures are very high. Furthermore, in order to cover large areas with reasonable accuracy, a high density of installed sensors are required [14], [15].

In modern navigation systems the dynamic properties of the traffic conditions are still not considered enough. Thus, in congested metropolises, the efficiency of route planning, which is particularly important in daily activities of public services such as police, fire brigades, medical emergency services, etc., is far from what is expected. Statistical traffic data is not sufficient in avoiding traffic jams and systems based on real-time data are needed [16], [17].

The use of reliable real-time urban traffic data can enable researchers and scientists to create traffic congestion control mechanisms which will help citizens by reducing their stress, time spent in traffic and energy consumption. However, providing low-cost, reliable and accurate traffic information has presented a major chal-

lenge in ITS [14].

A mobile crowd sensing paradigm represents an alternative solution to this challenge. With the advances in smartphones and tablet computers in terms of computing, communication and sensing power, citizens can deliver a distributed, viable and cost effective sensing infrastructure delivering real-time traffic information. Mobile crowd sensing therefore can be effectively used for the continuous measurement and monitoring of road and traffic conditions acquired from sensor equipped smart phones of the volunteered participants [14], [18], [19]. Smartphones come with a wealth of sensors which can provide more information than traditional infrastructural hardware because of users carrying these devices all the time [13], [20].

Manual information collection, requiring drivers' explicit interactions with the smart phones, can present safety issues. In the system described herein, the collection of data and warning the drivers are automated as much as possible. In order to provide congestion data, the drivers only need the mobile application running.

In this paper, Traffic and Road Monitoring System (TRMS), an urban traffic congestion control on route level and road monitoring system which exploits the power of participatory sensing and cloud messaging is proposed. Crowd intelligence is utilized within TRMS and the values for traffic congestion levels, arrival times, average speed on a given road segment are estimated by using the crowd-sensed data. TRMS provides proactive warnings and recommendations to drivers in the vicinity of, or on the route to, reported events. Moreover, the users of TRMS can report physical road conditions such as potholes, bumps or slippery or damaged road surface, and short-lived traffic events such as traffic accident, meeting, festival, celebration or protests. This would enable authorities to monitor traffic congestion and road conditions. Google Maps [21] are used to provide users with maps. Although Google Directions API [22] is used for direction finding between locations, TRMS uses its own algorithm to calculate travel time of a path where it is possible and advise its users in route planning.

The volunteers send traffic and road data using their smartphones that emanates from a number of sensors and are connected to the Internet through wireless networks. It is intended that the users are both providers and consumers of this data and this is fundamentally different from other systems which heavily rely on roadside equipment, obtrusive systems or the need of full cooperation from users.

Real-world experiments were conducted with a prototype implementation of the system and the results demonstrate system feasibility delivering accurate traffic estimation. The prototype system was also put into trial use and tested by 27 voluntary participants over a period of 58 days.

The contributions of our approach can be summarized as follows:

- Mobile Crowd Sensing (MCS) paradigm is used to estimate and calculate the traffic flow. Opportunistic and participatory sensing are both used.
- Cloud messaging is used to proactively warn users in near real-time. The proactive behavior is used to provide dynamic rerouting, dynamic driving instructions and notifications to drivers.
- The drivers also can report and get notified about physical road conditions and short-lived traffic events in their vicinity or on their route.

The subsequent sections of this paper are as follows: In Section II, related research is surveyed and compared to TRMS. Section III describes the TRMS; system architecture together with the algorithm used for calculating arrival time. In Section IV, TRMS is evaluated through a series of experiments and user trial. Finally, Section V draws conclusions from the work.

## II. RELATED WORK

In this section, related work similar to TRMS is reviewed. Traditionally, roadside equipment such as loop detectors and road side cameras are used in traffic prediction applications and provide only minimal information. Moreover, they are expensive to install and maintain [9]–[13] and they provide limited coverage [23].

Alternatively, smartphones with built-in sensors could act as sensors and provide fine-grained information because the users carry these devices all the time. Instead of installing roadside equipment, traffic flow and road condition data may be collected by smart phones resulting in reduced costs, avoidance of infrastructure maintenance, finer grained reporting and enhanced coverage.

A range of crowd sensing projects have sought to use smartphones'GPS sensors which were initiated to predict traffic congestions and traffic delays in addition to reporting road condition problems. Examples include MIT's CarTel [24] project, Microsoft's Nericell [25] project, Mobile Century and Mobile Millennium projects by UC Berkeley and Nokia [26], ParkNet [27], and CrowdITS [28] project in Queens University.

These projects have proven that GPS sensors on smartphones are feasible for sensing traffic flow, and even with a low penetration, we can get more detailed information and wider coverage than achievable with stationary sensors alone.

However none of these projects have addressed the use of cloud-assisted MCS for transportation and dynamic rerouting in case of a sudden congestion [13], [29].

Several commercial map and navigation services exist, such as Google Maps [21], Microsoft's Bing Maps [30] and Yandex Maps [31]. These services provide real-time traffic data, but they do not provide the proactive warning behavior supported by TRMS.

In order to effectively cross compare TRMS with pre-existing systems a number of functional attributes have been identified. These include:

- **Sensing method:** Options considered are crowd sensing, fixed (infrastructural) sensing and mobile crowd sensing.
- **User interaction:** Defines the manner in which users interact with the system. Options considered are participatory and/or opportunistic sensing. By the term participatory sensing, we mean that user intervention is needed in order to collect and send sensed data. Whereas, in opportunistic sensing, after the user initially grants permission, the smart phone sends the data autonomously without user intervention.
- **Proactive behavior:** Indicates if the system provides proactive warnings or recommendations to its users in the case of a sudden congestion. These recommendations might require dynamic rerouting.
- **Cloud computing usage:** Indicates if the system uses cloud assistance in its infrastructure.
- **Requirement for additional infrastructure support:** Considers if the system needs additional infrastructure to effectively operate.

The comparison of TRMS with related work is shown in Table 1. Although there are systems using different approaches, the related work listed in the table primarily covers mobile crowd sensing systems in the literature and systems which tend to use opportunistic user interactions.

Wan et al. [29] propose a mobile crowd sensing based system to predict traffic conditions and to avoid congestion. Experiments were carried out to verify the proposed approaches. Although this system uses MCS, cloud computing and also includes an incentive mechanism, it does not send proactive warnings nor recommendations to its users in case of a sudden congestion.

Predic and Stojanovic [14] also proposed the use of smartphones and crowd sensing in capturing dynamic and short-term traffic events like aggressive driving, risky behaviors, such as excessive braking, sudden lane changes, open turns, speed excess and route deviations. Geosocial network principles and proactive behavior are utilized in providing proactive notification and recommendations to drivers in the vicinity of reported events, while ensuring information reliability and preserving drivers' privacy. This system does not estimate traffic congestion nor does it embrace the power of cloud computing.

Yan et al. [13] do advocate a cloud-assisted MCS system for determining traffic conditions. Traffic recommendations and incentive mechanisms are provided, but the users are required to view the traffic condition on a map and then plan their route. Planning a route while driving might distract the driver and pose a serious life

threat. In addition, there are no proactive notifications by which to warn the user in case of a sudden congestion.

Liu et al. [23] present an opportunistic sensing based urban traffic monitoring system. This system facilitates crowd sensing and uses the bus riders' smartphones as source of the traffic information, in contrast to the existing works which heavily rely on intrusive sensing or full cooperation from their users. Public buses are exploited as dummy *probes* in order to detect road traffic conditions. The authors state that real-world experiments demonstrate the feasibility of their system, which achieves accurate and fine grained traffic estimation. An incentive mechanism is mentioned in their planned future work. The users of this system can only monitor traffic conditions. The system fails to provide path finding, proactive warnings or recommendations and does not utilize cloud messaging.

Hu et al. [32] present SmartRoad, a crowd-sourced road sensing system developed for detecting and identifying traffic regulators, including traffic lights, and stop signs. The detection and identification is realized without user intervention. SmartRoad acquires crowd sensed data from GPS sensors of the in-vehicle smartphones. It uses an incentive mechanism to motivate its users. The discovered traffic regulator information can be utilized in assisted-driving or navigation systems. A prototype system was implemented, and it was deployed on 35 external volunteer users' vehicles for a two month period. The authors state that the experiment results prove that SmartRoad is a successful system that can carry out the detection and identification tasks.

Lv et al. [33] present a novel deep-learning-based traffic flow prediction method, which considers spatial and temporal correlations. To learn generic traffic flow features, a stacked autoencoder model is used and this model is trained in a greedy layerwise fashion. This model works on sensor data acquired from a third-party database. The data is acquired from fixed sensors. It does not use MCS. It only predicts the future from the past data. For instance, if there is an unexpected situation such as an accident or a bad weather event, the system cannot adapt to such unforeseen circumstances. As it is only a prediction model, there is no proactive behavior nor cloud messaging.

Cerotti et al. [34] try to improve MCS by introducing an opportunistic and distributed MCS paradigm through the adoption of swarm intelligence. This approach is later used to implement route planning within an intelligent transportation system. A Markovian agent-based analytical modeling technique is used to evaluate the effectiveness of this approach. This system uses opportunistic MCS similar to our work, but it lacks proactive behavior and cloud messaging.

Dai et al. [35] suggest the Hybrid Spatio-Temporal Graph Convolutional Network (H-STGCN), for predicting future travel times from the current traffic flow.

**Table 1.** Comparison of TRMS with the other related work.

	Sensing method	User Interaction	Proactive behavior	Cloud assisted	Requires additional infrastructure
Wan et al. [29]	MCS	Opportunistic	✗	✓	Optional
Predic and Stojanovic [14]	MCS	Opportunistic	✓	✗	✗
Yan et al. [13]	MCS	Opportunistic	✗	✓	✗
Liu et al. [23]	MCS	Opportunistic	N/A	N/A	✗
Hu et al. [32]	MCS	Opportunistic	N/A	N/A	✗
Lv et al. [33]	Fixed sensors	N/A	N/A	N/A	✓
Cerotti et al. [34]	MCS	Opportunistic	✗	✗	✗
Dai et al. [35]	MCS	Opportunistic	✗	✓	✗
Kong et al. [36]	MCS, fixed sensors	Opportunistic	✗	✗	✗
Rahman et al. [37]	MCS, fixed sensors	Opportunistic	✗	✗	✓
TRMS	MCS	Opportunistic, participatory	✓	✓	✗

Similar to this work, the user synchronizes the current location with the cloud server in order to continuously update the travel time and the route if the user deviates from the planned one. It uses machine learning but lacks the proactive behavior.

Kong et al. [36] offer the Hierarchical Urban Anomaly Detection (HUAD) framework to detect urban anomalies based on spatio-temporal data. To improve the accuracy, spatio-temporal data from taxi and subway passengers are also integrated into the system. Their aim is to detect anomalies on time or predict anomalies in advance to prevent any loss of life or injuries. For taxi passengers, the data is acquired from GPS sensors within the taxis they are using. For the subway passengers, data is acquired from fixed sensors (card readers). This work utilizes mobile crowd sensing together with fixed sensors for data acquisition. There is no proactive behavior, route planning or real-time congestion warnings sent to users. Moreover, they did not use cloud computing.

Rahman et al. [37] report on their on-demand mobile crowd sensing approach to monitor the traffic status. It combines the capabilities of mobile devices and the cellular base stations for accurate real-time detection of traffic conditions. GPS sensors are not used for determining the location in this study. Instead, smartphones' location are calculated with the support of the cellular base stations which does not require the users' phone to send location updates. A deductive rule-based model is used for traffic condition estimation. This work lacks proactive behavior and cloud messaging. Furthermore, as a consequence of extensive use of cellular base stations, the battery consumption and network costs of the mobile device are reduced. The main drawback

experienced was increased system response time.

A cloud-assisted mobile crowd sensing system, Traffic and Road Monitoring System (TRMS), is proposed within this paper as a means of sensing traffic and road conditions. TRMS provides estimated arrival times and dynamic rerouting through proactive warnings/recommendations to the drivers in the vicinity of, or on the route to, reported events. TRMS is novel and offers advancement beyond previously reviewed related work by employing a combination of (1) opportunistic sensing paradigm, which is used in sensing traffic flow, and (2) participatory sensing, which is utilized for reporting physical road conditions and short-term events. Moreover, TRMS makes it possible for authorities to monitor traffic congestion flow and road conditions. Real-world experiments have been conducted with a prototype implementation of the system. The results demonstrate the feasibility of this system which achieves accurate traffic estimation and journey time.

### III. TRAFFIC AND ROAD MONITORING SYSTEM (TRMS)

In this section, we provide a motivating scenario demonstrating the benefits of the proposed system in real world use. In Section III-A, the trial use of the prototype is described, while Section III-B explains the system architecture. Finally Section III-C describes the approach to the calculation of traffic congestion levels and the algorithm used for calculating travel times.

#### A. MOTIVATION

When a driver, who wants to travel from his current location to a destination, submits his/her preferred destination location to the TRMS mobile application

running on his/her smartphone, the application communicates with the server and shows the route that will take the least amount of time while taking into account the user's preferences. The congested parts of the roads are also shown to the driver who is expected to approve the route. While driving on the chosen route, if the mobile application receives a notification about an event on the route, it deduces that the situation has changed as a result of a recent event. Therefore, now there is another route with a lower driving time available. While the mobile application warns the driver immediately via visual and audio means, the driver does not need to interact with the smartphone. The application proactively deduces if the driver approves the new route or not. The driver approves the new route and as a result, the travel duration is reduced. The system proposed in this paper is capable of realizing this scenario, which shows the advantage of using this system.

## B. SYSTEM ARCHITECTURE

The system architecture of TRMS shown in Fig. 1. It is a cloud-assisted MCS architecture that consists of mobile application components, server components and visualization components. These components are organized in a distributed architecture in which the *opportunistic* and *participatory* crowd sensing co-exist. The data collection can be performed with active user involvement which is termed as participatory sensing, or via opportunistic sensing which does not necessitate any active user involvement.

In TRMS, opportunistic crowd sensing is used for traffic congestion control; whereas participatory sensing is commissioned for the reporting of physical road conditions and short-term events such as; meetings, festivals or protests.

The client side may include smartphones, tablets, laptops and desktop computers. The mobile application runs on the smartphone. In addition to the mobile application, the users of TRMS can use the system through internet browsers. The mobile client application continuously sends navigation data to the server and the server stores this data. The navigation data consists of location in latitudes and longitudes, velocity (speed and direction) and time. The frequency of this is adjusted according to the users' speed. As the user's speed increases, the frequency of navigation data update is also increased to get the fine-grained data.

The route planner component is responsible for constructing a route for going from one point to a destination point based on the user preferences (i.e. shortest path, shortest arrival time, etc.). It proactively monitors the active route and it may change the route based on the newly arrived congestion data. This situation is notified to the user immediately.

In order to motivate user participation in crowd sensing activities, an incentive mechanism is incorporated; since without one, most of the users might tend to consume traffic data without participating. A credit-based incentive mechanism is utilized. As long as a user uploads data, he/she can consume. If the user has enough credits, he/she does not need to upload, until the user has no credits left. This functionality is useful if the communication costs are high at the time or the smart phone's battery is low. The incentive component is responsible for storing and managing user credits.

The users can report short-lasting traffic events (such as meeting, festival, celebration, protest and etc.) and also physical road defects (such as slippery road, damaged road, potholes, bumps and etc) to the server. The reporting component is responsible for these tasks respectively.

The implemented prototype system provides a common RESTful (Representational State Transfer) API (Application Programming Interface). Thus, it supports multiple client platforms and devices. The mobile application is designed to be used on smartphones and any internet browser can be used to reach the TRMS from desktop computers. In order to impose security and privacy, authentication and authorization mechanisms are used. The OAuth 2.0 [38] protocol is used for authorization. The key services for the system are identified and abstracted from the internal implementation. The API calls are implemented through HTTP method calls such as get, post, update, and delete. Adding new services can be easily accomplished without affecting the previous services.

The notification component is responsible for sending push notifications through Firebase Cloud Messaging (FCM) [39] service. Additionally, clients can send data directly to the server using the RESTful API. The warnings and recommendations are sent to smartphone users via a push notification service. The advantage of push notifications is that the user gets the message almost instantly when there is an internet connection. Implementing a push notification mechanism for mobile devices is a challenge because they do not have consistent Internet connections all the time and they have limited resources. Firebase Cloud Messaging (FCM) enables the addressing of this problem as it is a free service which enables developers to send messages between servers and client apps. FCM supports the storing of messages when a receiver is not connected to the Internet. Thus, the burden of cloud messaging is transferred to an external server for free through the use of FCM.

The location is acquired from the GPS sensor of the smartphone. Although the GPS sensor has high power consumption and it drains the smartphone's battery quickly, the users can recharge in the vehicle by using car chargers. Thus, this should not be a problem.

MongoDB is used for storing continuous sensor

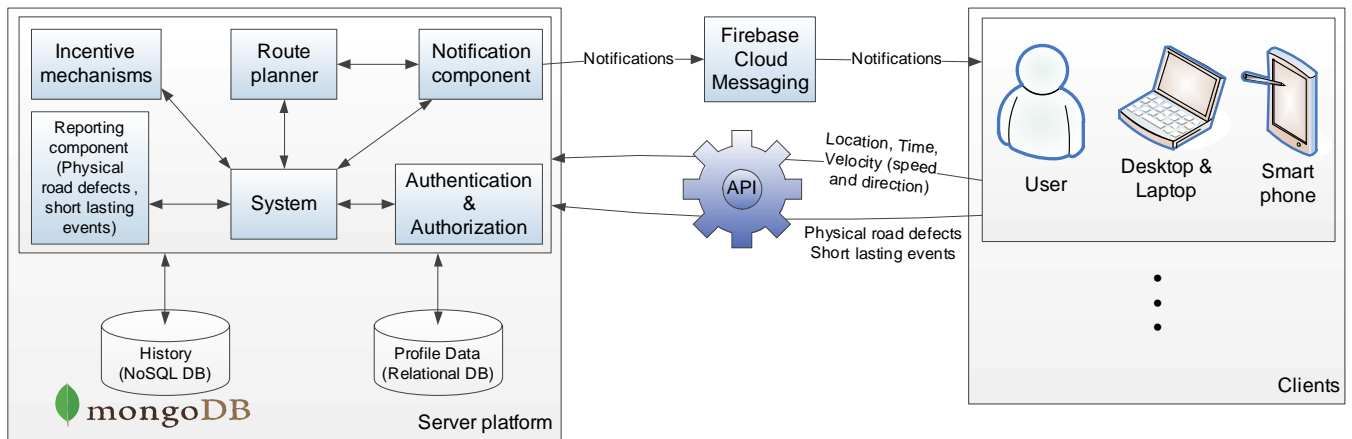


Figure 1. System architecture of TRMS.

data that are composed of coordinates of geographic locations, time and velocity. MongoDB, which is a document-oriented NoSQL database, was chosen because it is secure, scalable, and flexible. It provides a rich set of tools and features [40], [41]. Other information such as the user profile and user reported events are stored in a relational database. The mobile client application is developed for the Android platform [42], Google Maps [21] is used to present users with maps, and Google Directions API [22] provides path finding between two locations. The TRMS' algorithm for calculating traffic congestion levels and the travel times is explained in Section III-C. The client application shows the estimated arrival time, duration of the journey, speed and distance to the driver as shown in Fig. 2. The geographic position, heading direction, and the route are shown visually to the user on the map. This is the standard information shown to the user on dedicated GPS receivers. Fig. 3 shows the mobile application in the free roaming mode, where the user does not enter a destination. In this mode, only the speed, geographic position, and heading direction are shown to the user. TRMS also monitors the possible routes that might be taken by the driver (when there is a fork in the road) and if there is congestion, the server proactively and autonomously notifies the client about the situation. For example, in Fig. 3, the user is warned that the left route has become congested without any user intervention.

In Fig. 4, the user is warned about a road construction event. When the mobile application receives an event notification, it warns the driver by audio and text messages. As more drivers react by voting for the event, this situation is emphasized to the driver by using different colors and sounds. The color turns from orange to red as the number of users, which agree with the presented data, increases. Thus, the quality of data is assured and

trust is provided.

Fig. 5 shows a user confirmation of a traffic accident event. The date of the first notification of this event and the number of people who confirmed the data collected regarding this event are also shown to the user. Users can report events and confirm/refute events after the commute, provided that the event that took place was on their route. This mechanism provides a group check and group management of reported events with consensus emerging as a powerful instrument in ensuring the integrity and quality of reported events. During driving, the driver can double tap the screen of the device to put a bookmark on the current location. Later on, after the trip or commute, the driver can enter the details of the event. The reason for this is not to distract the driver while driving and to prevent any dangerous situations.

### C. TRAFFIC CONGESTION CALCULATION

As stated previously, the proposed system calculates congestion levels from the crowd sensed data. It is hard to store congestion data embedded within the maps, because maps and directions are acquired from external sources. As a solution to this problem, a lazy congestion detection approach is employed. In this approach, when there is a need to determine congestion levels, the congestion for the routes is calculated instantly using the crowd sensed data, and caching is used to increase performance. When there is a congestion request for a route, the cache is searched first for a complete or a partial match. If there is a partial match, then only the congestion information for non-matching parts of the route are calculated.

The pseudocode of the algorithm for calculating travel time of a specific route, which is composed of a set of points, is given in Fig. 6. This is a simplified version of the actual algorithm, which considers only one probe (data producer). In the implemented version of TRMS,



Figure 2. The navigation view of the TRMS mobile application.

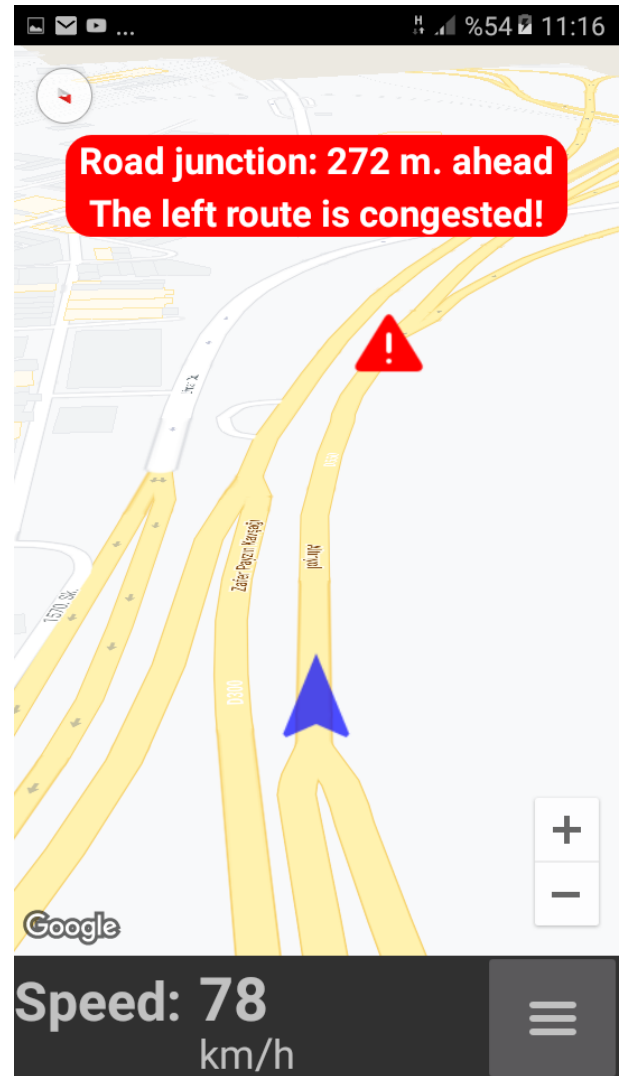


Figure 3. The congestion warning in free roaming mode.

if there are multiple probes, the average of all of the probes or the fastest probe is considered based on the user preferences. In the algorithm, points in the route are iterated through. The *findNearestGeoLoc* function searches crowd sensed data in the database to match if there is a recorded point of a route. A route object is passed to this function, because the heading direction must also be considered so as not to proceed in the opposite direction.

If there are no matching points, then the algorithm iterates the other points in the route until it finds a matching point. Duration for these non-matched points is acquired by using *calcDurationBOM* function which tries to obtain the duration value for the partial route. This function initially uses the past crowd sensed data stored for the same day of week and the same hour to obtain the duration value. If this fails, it obtains the duration from Google Directions service.

On the other hand, if a sequence of points matched, then the algorithm calculates the duration from the timestamps of the starting point and the ending point. The *contains* function checks if a given point is indeed on the route. Subsequently, the index of the ending point's reciprocal point is obtained by calling *findNearestIndex* function. The counter is assigned to this index point in order to skip the other points in the sequence. These operations are repeated until all of the points in the route are iterated. The average speed of a road segment is calculated by dividing the distance by the travel time and the congestion levels are derived from the road segment's average speed via a discrete mapping function.

#### IV. EXPERIMENT AND TRIAL USE

Detailed experimentation was undertaken together with a user trial. Section IV-A, IV-B and IV-C provide details

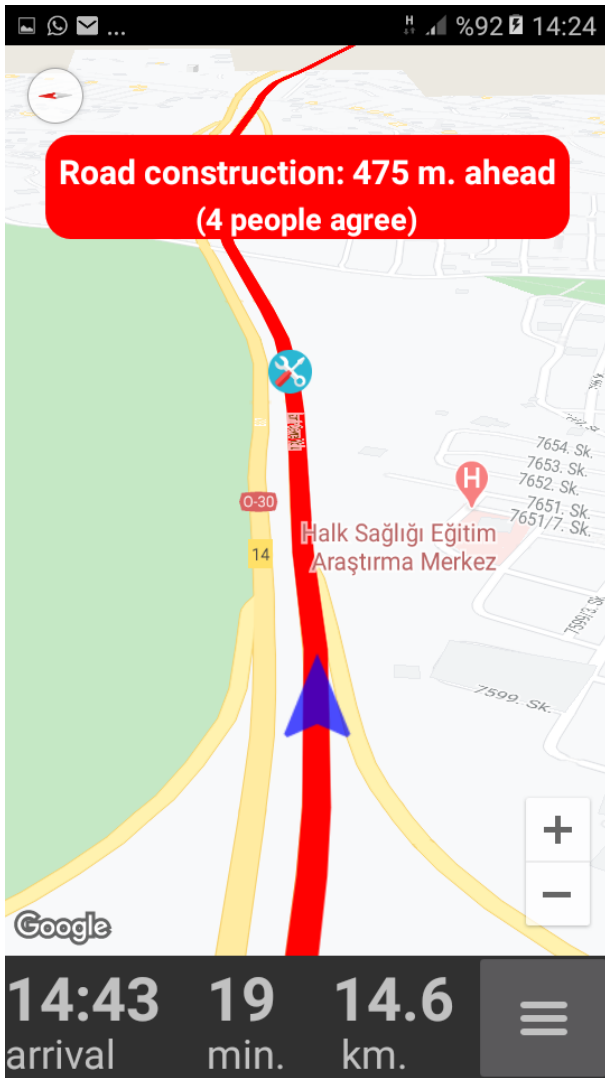


Figure 4. The user is warned about a construction event.

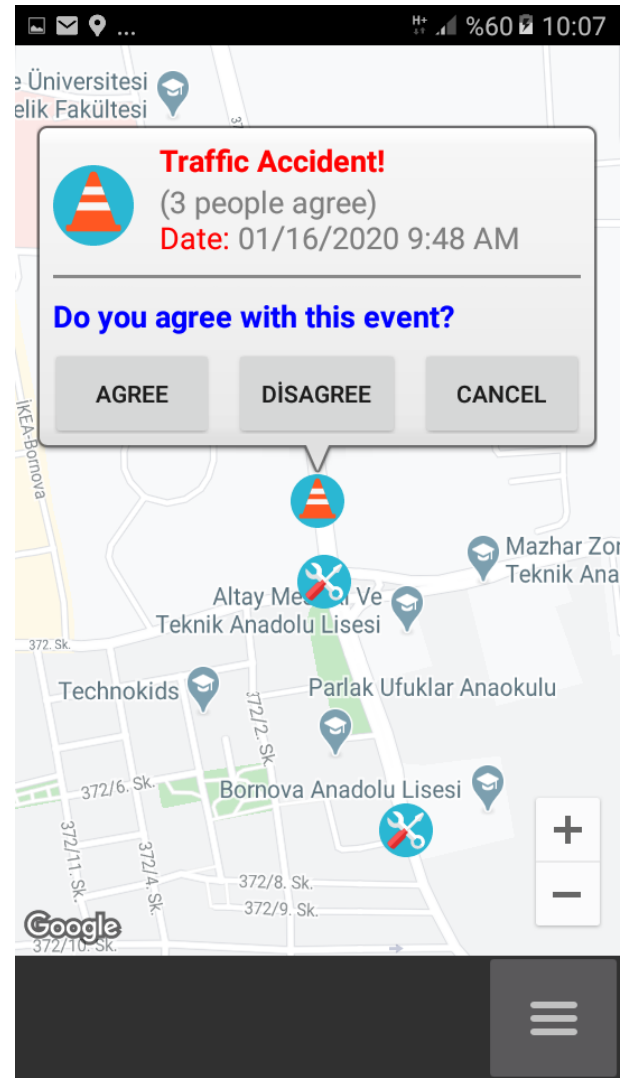


Figure 5. Voting for a reported traffic accident.

of the experiments, associated results and user trials respectively. Evaluating the incentive and notification mechanisms is part of the future work.

#### A. EXPERIMENT

Systematic experiments were undertaken in order to evaluate the performance of TRMS in estimating traffic congestion and warning its users in real-world conditions. The experiment stage was divided into two phases. In phase one, a shorter route with two alternative paths was chosen. In phase two, a longer route with three alternative paths was chosen. In phase one, traffic flow estimation was evaluated. In phase two, in addition to traffic flow estimation, dynamic rerouting capability of the system was also evaluated. The experiment setup deployed in one of the cars is shown in Fig 7. All of the routes were located within an urban environment, the city of İzmir in Turkey, with heavy traffic volumes.

In phase one, two alternative routes were selected in order to go from a source location (latitude: 38.441049, longitude: 27.169812) to a destination location (latitude: 38.424159, longitude: 27.137666) as shown in Fig. 8. The routes are named as Route A (blue) and Route B (green). Route A and route B are 4.1 and 3.6 kilometers long respectively. Only these two routes were included to simplify the experiment and to reduce resources needed to carry out the experiment. The system was configured to discard other routes with unknown traffic data during the experiment to ensure that only one of the two routes would be selected and the selected route would be followed until reaching the destination location.

Four cars were used in phase one of the experiment. First two cars act as probes sending location data continuously, each taking one of the two routes. There is a finish-to-start relation between the journeys of the probe cars and the third and fourth cars. After the probe



**Input:** *route* object consisting of geopoints  
**Output:** travel time *dur*

```

i ← 0
dur ← 0
while i < route.size() do
    geoloc ← route.get(i)
    start ← findNearestGeoLoc(geoloc, route)
    j ← i
    while (start = null) do
        j ← j + 1
        geoloc ← route.get(j)
        start ← findNearestGeoLoc(geoloc, route)
    end while
    dur ← dur + calcDurationBOM(route, i, j - 1)
    end ← start
    while route.contains(end.nextGeoLoc()) do
        end ← end.nextGeoLoc()
    end while
    dur ← dur + start.timeElapsed(end)
    i ← route.findNearestIndex(end)
    i ← i + 1
end while

```

Figure 6. The algorithm for calculating travel time of a specific route.

cars arrive at the destination, the third car starts its journey and takes the route suggested by the system. This route is selected according to the data acquired from the probe cars based on the shortest arrival time. The fourth car takes the other route to find out which route has the shortest arrival time.

In phase two, the experiment setup is more complicated, not only because there are three alternative paths but dynamic rerouting function of the system is evaluated. Similar to phase one, the drivers try to go from a source location (latitude: 38.488515, longitude: 27.028715) to a destination location (latitude: 38.458472, longitude: 27.125687) as shown in Fig. 9. The routes are 12.8, 11.1 and 13.2 kilometers long.

Nine cars were used for phase two. First set of three cars act as probes sending location data continuously, each taking one of the three routes. Second set of three cars also act as probes. They begin their journeys when the first set of cars reach halfway of their journeys. After one of the second set of cars reach halfway, seventh car begins its journey and takes the route having shortest arrival time suggested by the system. By using two sets of probe cars, more up-to-date data is acquired. We can also test dynamic rerouting because the route with the shortest arrival time might change as data from the second set of cars are acquired. The participants were instructed to follow the suggestions provided by the system. Thus, if the route changes as a result of dynamic rerouting, then the participants will follow the new route. The last two cars follow the seventh car and

Table 2. Experiment results of phase one.

Experiment no.	Suggested route	Travel duration (minutes)		Success
		Route A	Route B	
1	B	12.2	7.8	✓
2	B	9.6	7.2	✓
3	A	11.7	12.3	✓
4	A	9.2	9.0	✗
5	B	9.6	8.3	✓
6	A	12.1	12.9	✓
7	B	9.8	10.4	✗
8	B	13.3	9.6	✓
9	B	12.2	8.8	✓
10	A	12.3	17.4	✓
11	B	10.8	8.9	✓
12	A	11.3	12.2	✓

they take the routes not taken by the seventh car to measure travel time of the other routes to be used for comparison.

## B. RESULTS AND DISCUSSION

The experiments of phase one was repeated 12 times and the results are shown in Table 2. All of the experiments were conducted during rush hours (working days of the week, between 7:30-9.00 and between 17:00-18:30) and as a result, the success rate of the system in predicting the route with the shortest arrival time was 83%. The system's suggestion was wrong in only two cases, numbered 4 and 7, where the travel duration for the two routes were close (less than one minute). This error was due to the finish-to-start relation between the journeys of the probe cars and the third car. As a result of this, some time passes and the traffic conditions may change. A faster route may become slower.

The data sent from the probe cars and the travel durations of the routes for each experiment were verified. The route suggestions based on this data were also correct.

Without the traffic congestion estimation, the shorter route (route B) would be chosen every time. However, in experiments numbered 3, 6, 10 and 12, there was congestion in route B and route A was faster and had a shorter arrival time. As a result, route A was chosen and the travel duration was reduced, showing the benefit of mobile crowd sensed traffic estimation.

The experiments of phase two was repeated 14 times and the results are shown in Table 3. All of the experiments were conducted during working days of the week, between 7:58 and 19.53 and as a result, the success rate of the system in predicting the route with the shortest arrival time was 86%. The system's suggestion was

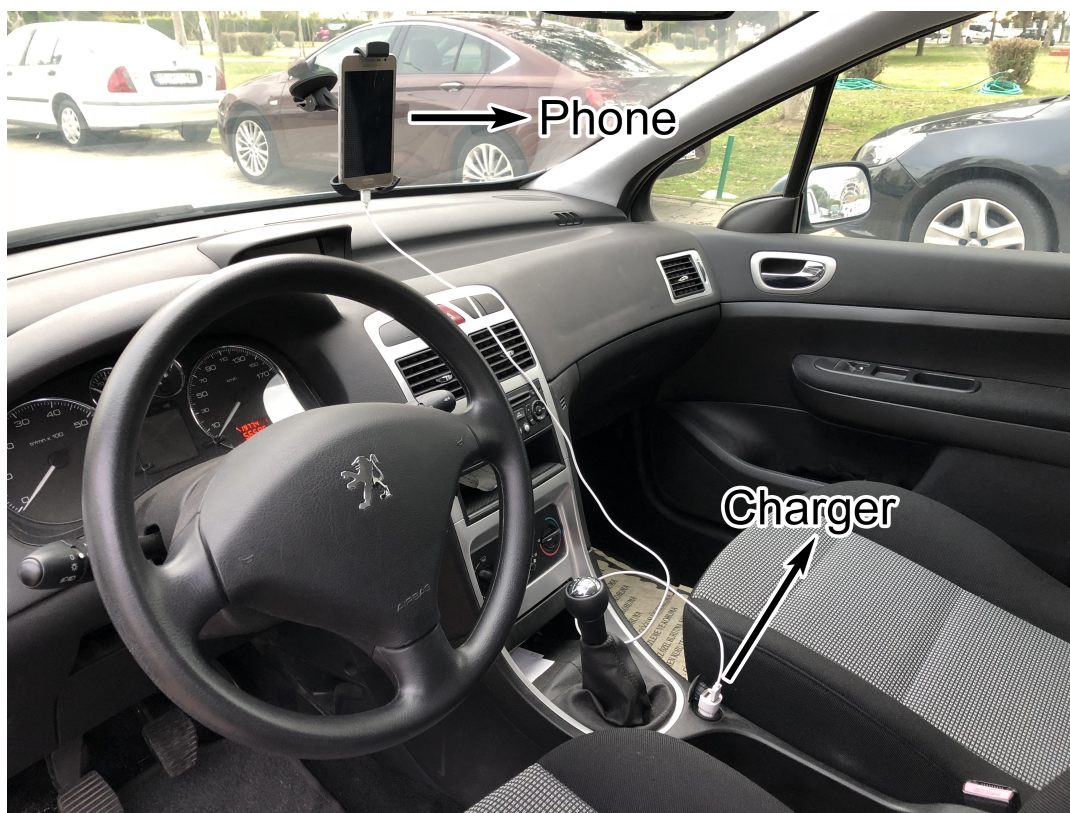


Figure 7. The experiment setup deployed in a car.

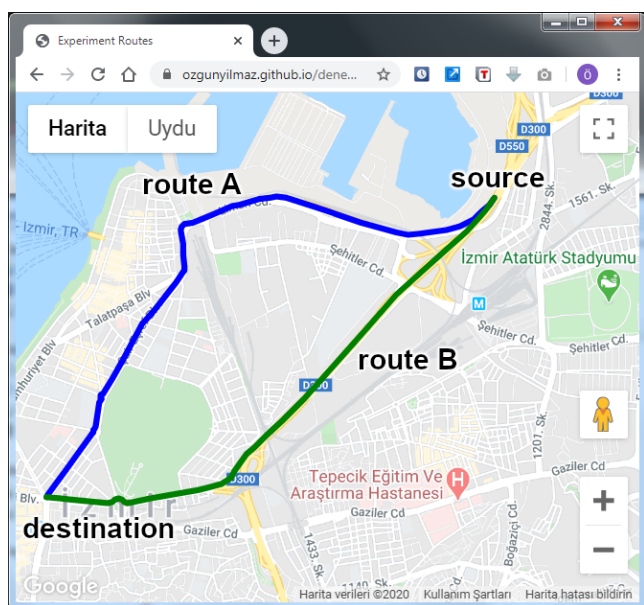


Figure 8. Selected alternate routes for the experiment.

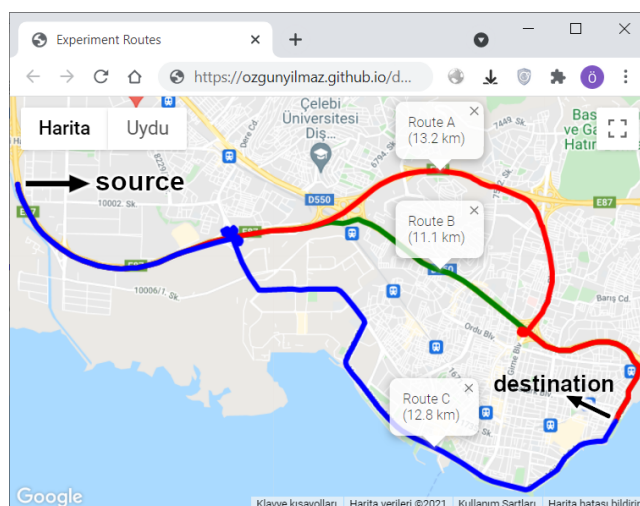


Figure 9. Selected alternate routes for the experiment.

wrong in only two cases, numbered 3 and 8. During the cases numbered 4, 9 and 12, dynamic rerouting occurred. This transition is indicated in suggested route column of

Table 3 by using right arrow. While the left-hand side of the arrow indicates the suggested road in the beginning, the right-hand side indicates the updated route as a result of dynamic rerouting. Experiments conducted in phase two shows the advantage of dynamic rerouting, since it increased the success rate of the system and

**Table 3.** Experiment results of phase two.

Exp. no.	Suggested route	Travel duration (minutes)			Success
		Route A	Route B	Route C	
1	A	19.2	36.7	23.6	✓
2	B	13.7	12.1	17.6	✓
3	A	13.3	11.8	18.5	✗
4	A → B	14.6	13.2	18.9	✓
5	A	18.3	24.7	22.4	✓
6	C	29.4	38.9	22.3	✓
7	A	13.9	18.1	16.8	✓
8	A	14.2	13.7	17.8	✗
9	A → B	19.6	18.0	22.3	✓
10	B	23.5	17.9	22.4	✓
11	C	30.3	36.7	23.6	✓
12	B → A	13.8	15.1	17.5	✓
13	B	14.3	12.6	19.2	✓
14	B	24.9	20.8	26.7	✓

without it there would be three more wrong cases and the success rate would drop to 71%.

### C. TRIAL USE

In addition to the experiments, the prototype system was put into trial use to test other aspects of the system, such as gathering crowd sensed data about physical road conditions and short-term events. It is used by 27 voluntary participants for 58 days in the city of İzmir. During this trial use, approximately 38,481 km and 742 driving hours of traffic road data were collected. At the end of the trial use, 12 traffic accidents, 3 road constructions and 43 physical road defects (damaged road, potholes, bumps and etc.) were reported by the test subjects. The road defects were reported to the corresponding authorities.

Users of TRMS can view real-time traffic condition on maps. Fig. 10 shows a real-time traffic congestion map captured during the trial use. The congestion levels encoded in different colors are listed in the legend. To evaluate this map, it is compared with Google Maps which shows traffic condition. Google Maps serves as an open, independent, transparent and publicly-available baseline and can facilitate comparisons. Google defines four levels of traffic congestion in color codes and keeps speed thresholds of the color codes secret. Thus, we can only compare travel times and make a visual comparison. While TRMS estimated the travel duration for route A shown in Fig 10 as 38 minutes, Google Maps' estimate was 36 minutes. And for route B, while our system estimated the travel duration for route A shown in Fig 10 as 32 minutes, Google Maps' estimate was 33 minutes.

To make a visual comparison the traffic conditions from the two maps were extracted and then superimposed using an image editing software as shown in Fig 11. The thicker line shows the traffic condition obtained by our system, whereas the thinner line shows the traffic condition from Google Maps. After viewing the superimposed color encoded traffic conditions, one can see that the both predictions are similar. The beginning and ending points of the colored paths were slightly shifted, but they mostly overlap. This shift is probably the result of different threshold values.

### V. CONCLUSION

In this paper the design, implementation, and evaluation of TRMS are presented. TRMS is a cloud assisted mobile crowd sensing system for road monitoring and traffic congestion control at route level. The volunteered users send traffic and road data using their smartphones with the help of built-in sensors and Internet connection. TRMS then uses this crowd sourced traffic information to make arrival time and route planning calculations within its own algorithm. This is in contrast to other systems which rely heavily on roadside equipment, obtrusive systems or need full cooperation from users. The main contribution of TRMS is supplying proactive warnings and recommendations to drivers in the proximity of or on the route to reported events. This proactive behaviour is not supported by mainstream commercial products on the market. Moreover, the authorities and the users are able to monitor traffic congestion and road conditions by using TRMS. Real-world experiments were undertaken with a prototype implementation of the system. The results demonstrate the efficiency of this system which achieves accurate traffic estimation. The prototype system was also put into trial use and the corresponding authorities were notified about the detected road defects.

As future work, we will test and evaluate effectiveness of the incentive and notification mechanisms which are subsystems of TRMS.

### References

- [1] Y. Tian, W. Hu, B. Du, S. Hu, C. Nie, and C. Zhang, "Iqga: A route selection method based on quantum genetic algorithm-toward urban traffic management under big data environment," *World Wide Web*, vol. 22, no. 5, pp. 2129–2151, Sep. 2019. [Online]. Available: <https://doi.org/10.1007/s11280-018-0594-x>
- [2] X. Kong, X. Song, F. Xia, H. Guo, J. Wang, and A. Tolba, "Lotad: long-term traffic anomaly detection based on crowdsourced bus trajectory data," *World Wide Web*, vol. 21, no. 3, pp. 825–847, May 2018. [Online]. Available: <https://doi.org/10.1007/s11280-017-0487-4>
- [3] T. Khalid, A. N. Khan, M. Ali, A. Adeel, A. ur Rehman Khan, and J. Shuja, "A fog-based security framework for intelligent traffic light control system," *Multimed. Tools Appl.*, vol. 78, pp. 24 595–24 615, 2019. [Online]. Available: <https://doi.org/10.1007/s11042-018-7008-z>
- [4] A. Essien, I. Petrounias, P. Sampaio, and S. Sampaio, "A deep-learning model for urban traffic flow prediction with traffic

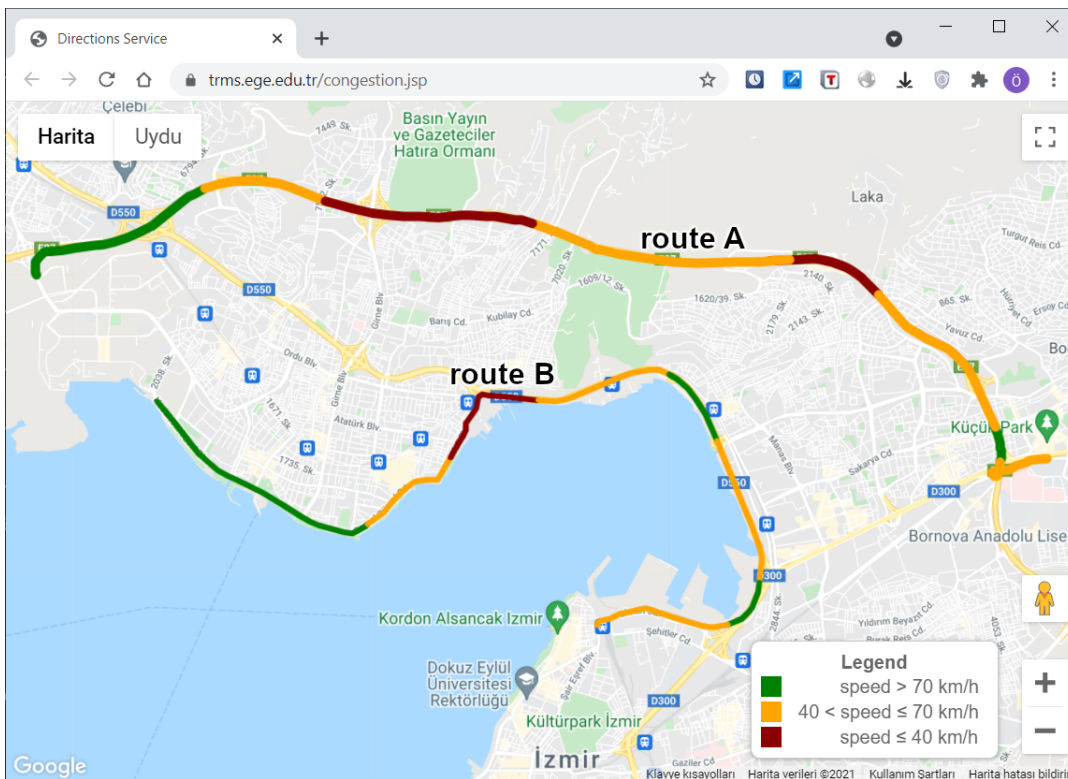


Figure 10. Real-time traffic congestion map.

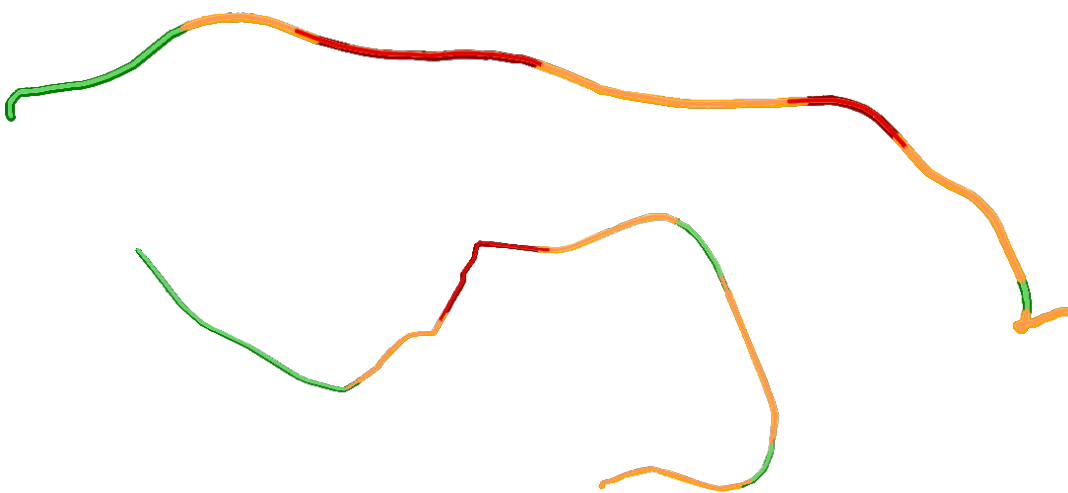


Figure 11. Real-time traffic congestion map.

events mined from twitter,” *World Wide Web*, Mar. 2020. [Online]. Available: <https://doi.org/10.1007/s11280-020-00800-3>

[5] D. He, S. Wang, B. Ruan, B. Zheng, and X. Zhou, “Efficient and robust data augmentation for trajectory analytics: a similarity-based approach,” *World Wide Web*, vol. 23, no. 1, pp. 361–387, Jan. 2020. [Online]. Available: <https://doi.org/10.1007/s11280-019-00695-9>

[6] K. Pholsena, L. Pan, and Z. Zheng, “Mode decomposition based deep learning model for multi-section traffic prediction,” *World Wide Web*, vol. 23, no. 4, pp. 2513–2527, Jul. 2020. [Online]. Available: <https://doi.org/10.1007/s11280-020-00791-1>

[7] H.-Y. Cheng, V. Gau, C.-W. Huang, and J.-N. Hwang, “Advanced formation and delivery of traffic information in intelligent transportation systems,” *Expert. Syst. Appl.*, vol. 39, no. 9, pp. 8356–8368, 2012. [Online]. Available: <https://doi.org/10.1016/j.eswa.2012.01.184>

[8] Y. Li and D. Chen, “A learning-based comprehensive evaluation model for traffic data quality in intelligent transportation systems,” *Multimed. Tools Appl.*, vol. 75, pp. 11 683–11 698, 2016. [Online]. Available: <https://doi.org/10.1007/s11042-015-2676-4>

[9] M. Chen, V. Leung, S. Mao, Y. Xiao, and I. Chlamtac, “Hybrid geographical routing for flexible energy-delay tradeoffs,” *IEEE*

- Trans. Veh. Technol.*, vol. 58, no. 9, pp. 4976–4988, 2009. [Online]. Available: <https://doi.org/10.1109/TVT.2009.2025767>
- [10] M. Chen, D. Mau, Y. Zhang, T. Taleb, and V. Leung, “Vendnet: vehicular named data network,” *Veh. Commun.*, vol. 1, no. 4, pp. 208–213, 2014. [Online]. Available: <https://doi.org/10.1016/j.vehcom.2014.09.002>
- [11] J. Liu, J. Wan, Q. Wang, P. Deng, K. Zhou, and Y. Qiao, “A survey on position-based routing for vehicular ad hoc networks,” *Telecommun. Syst.*, vol. 62, no. 1, pp. 15–30, 2016. [Online]. Available: <https://doi.org/10.1007/s11235-015-9979-7>
- [12] D. Zhang, J. Wan, Z. He, S. Zhao, K. Fan, S. O. Park, and Z. Jiang, “Identifying region-wide functions using urban taxicab trajectories,” *ACM Trans. Embed. Comput. Syst.*, vol. 15, no. 2, pp. 1–19, 2016. [Online]. Available: <https://doi.org/10.1145/2821507>
- [13] H. Yan, Q. Hua, D. Zhang, J. Wan, S. Rho, and H. Song, “Cloud-assisted mobile crowd sensing for traffic congestion control,” *Mobile Netw. Appl.*, vol. 22, no. 6, pp. 1212–1218, apr 2017. [Online]. Available: <https://doi.org/10.1007/s11036-017-0873-2>
- [14] B. Predic and D. Stojanovic, “Enhancing driver situational awareness through crowd intelligence,” *Expert. Syst. Appl.*, vol. 42, no. 11, pp. 4892–4909, jul 2015. [Online]. Available: <https://doi.org/10.1016/j.eswa.2015.02.013>
- [15] D. Stojanovic, B. Predic, and N. Stojanovic, “Mobile crowd sensing for smart urban mobility,” in *European Handbook of Crowdsourced Geographic Information*. Ubiquity Press London, UK, 2016, vol. 371.
- [16] W. Chmiel, J. Dańda, A. Dziech, S. Ernst, P. Kadłuczka, Z. Mikrut, P. Pawlik, P. Szwed, and I. Wojnicki, “INSIGMA: an intelligent transportation system for urban mobility enhancement,” *Multimedia Tools and Applications*, vol. 75, no. 17, pp. 10 529–10 560, 2016. [Online]. Available: <https://doi.org/10.1007/s11042-016-3367-5>
- [17] W. Chmiel, I. Skalna, and S. Jędrusik, “Intelligent route planning system based on interval computing,” *Multimedia Tools and Applications*, vol. 78, no. 4, pp. 4693–4721, 2019. [Online]. Available: <https://doi.org/10.1007/s11042-018-6714-x>
- [18] S. Giordano and D. Puccinelli, “When sensing goes pervasive,” *Pervasive Mob. Comput.*, vol. 17, pp. 175–183, 2015. [Online]. Available: <https://doi.org/10.1016/j.pmcj.2014.09.008>
- [19] D. Cerotti, S. Distefano, G. Merlino, and A. Puliafito, “A crowd-cooperative approach for intelligent transportation systems,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–11, 2016. [Online]. Available: <https://doi.org/10.1109/TITS.2016.2609606>
- [20] Y. Wang, Z. Yan, W. Feng, and S. Liu, “Privacy protection in mobile crowd sensing: a survey,” *World Wide Web*, vol. 23, no. 1, pp. 421–452, 2020. [Online]. Available: <https://doi.org/10.1007/s11280-019-00745-2>
- [21] Google, “Google maps platform,” Accessed: Jul. 20, 2021. [Online]. Available: <https://developers.google.com/maps/documentation>
- [22] Google, “Directions API,” Accessed: Jul. 20, 2021. [Online]. Available: <https://developers.google.com/maps/documentation/directions/>
- [23] Z. Liu, S. Jiang, P. Zhou, and M. Li, “A participatory urban traffic monitoring system: The power of bus riders,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2851–2864, oct 2017. [Online]. Available: <https://doi.org/10.1109/TITS.2017.2650215>
- [24] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden, “Cartel: a distributed mobile sensor computing system,” in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, SenSys 2006, Boulder, Colorado, USA, October 31 - November 3, 2006*, A. T. Campbell, P. Bonnet, and J. S. Heidemann, Eds. ACM, 2006, pp. 125–138.
- [25] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “Nericell: rich monitoring of road and traffic conditions using mobile smartphones,” in *Proceedings of the 6th International Conference on Embedded Networked Sensor Systems, SenSys 2008, Raleigh, NC, USA, November 5-7, 2008*, T. F. Abdelzaher, M. Martonosi, and A. Wolisz, Eds. ACM, 2008, pp. 323–336.
- [26] J. C. Herrera, D. B. Work, R. Herring, X. J. Ban, Q. Jacobson, and A. M. Bayen, “Evaluation of traffic data obtained via gps-enabled mobile phones: The mobile century field experiment,” *Transp. Res. Part C Emerg. Technol.*, vol. 18, no. 4, pp. 568–583, 2010. [Online]. Available: <https://doi.org/10.1016/j.trc.2009.10.006>
- [27] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe, “Parknet: Drive-by sensing of road-side parking statistics,” in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys ’10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 123–136.
- [28] K. Ali, D. Al-Yaseen, A. Ejaz, T. Javed, and H. S. Hasanein, “Crowdits: Crowdsourcing in intelligent transportation systems,” in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, April 2012, pp. 3307–3311.
- [29] J. Wan, J. Liu, Z. Shao, A. Vasilakos, M. Imran, and K. Zhou, “Mobile crowd sensing for traffic prediction in internet of vehicles,” *Sensors*, vol. 16, no. 1, p. 88, jan 2016. [Online]. Available: <https://doi.org/10.3390/s16010088>
- [30] Microsoft, “Bing maps - directions, trip planning, traffic cameras & more,” Accessed: Jul. 20, 2021. [Online]. Available: <https://bing.com/maps>
- [31] Yandex, “Yandex.maps: search for places, addresses, and public transport,” Accessed: Jul. 20, 2021. [Online]. Available: <https://yandex.com/maps/>
- [32] S. Hu, L. Su, H. Liu, H. Wang, and T. F. Abdelzaher, “Smartroad: Smartphone-based crowd sensing for traffic regulator detection and identification,” *ACM Trans. Sens. Netw.*, vol. 11, no. 4, pp. 1–27, 2015. [Online]. Available: <https://doi.org/10.1145/2770876>
- [33] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, “Traffic flow prediction with big data: A deep learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.
- [34] D. Cerotti, S. Distefano, G. Merlino, and A. Puliafito, “A crowd-cooperative approach for intelligent transportation systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 6, pp. 1529–1539, 2017.
- [35] R. Dai, S. Xu, Q. Gu, C. Ji, and K. Liu, “Hybrid spatio-temporal graph convolutional network: Improving traffic prediction with navigation data,” in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD ’20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 3074–3082. [Online]. Available: <https://doi.org/10.1145/3394486.3403358>
- [36] X. Kong, H. Gao, O. Alfarraj, Q. Ni, C. Zheng, and G. Shen, “Huad: Hierarchical urban anomaly detection based on spatio-temporal data,” *IEEE Access*, vol. 8, pp. 26 573–26 582, 2020.
- [37] S. Abdul Rahman, A. Mourad, and M. El Barachi, “An infrastructure-assisted crowdsensing approach for on-demand traffic condition estimation,” *IEEE Access*, vol. 7, pp. 163 323–163 340, 2019.
- [38] OAuth, “OAuth 2.0,” Accessed: Jul. 20, 2021. [Online]. Available: <https://oauth.net/2/>
- [39] FCM, “Firebase cloud messaging,” Accessed: Jul. 20, 2021. [Online]. Available: <https://firebase.google.com/>
- [40] MongoDB, “MongoDB architecture,” Accessed: Jul. 20, 2021. [Online]. Available: <https://www.mongodb.com/mongodb-architecture>
- [41] S. Parikh and K. Stirman, “Schema design for time series data in MongoDB,” Dec. 2014, Accessed: Jul. 20, 2021. [Online]. Available: <https://www.mongodb.com/blog/post/schema-design-for-time-series-data-in-mongodb>
- [42] Google, “Android developers,” Accessed: Jul. 20, 2021. [Online]. Available: <https://developer.android.com/>



things and multi-agent systems.

**ÖZGÜN YILMAZ** is an Associate Professor in the Department of Computer Engineering at Ege University, İzmir, Turkey. He received his M.Sc. and Ph.D. degrees in computer engineering in the areas of multi-agent systems, semantic web and artificial intelligence from Ege University in 2009 and 2013 respectively. His research interests include mobile computing, artificial intelligence, ubiquitous computing, internet of



networks, agent-oriented programming and ubiquitous computing.

**LEVENT GÖRGÜ** is a Research Scientist in the School of Computer Science at University College Dublin (UCD). He received his M.Sc. and Ph.D. degrees in Computer Science in the areas of ubiquitous computing, service discovery and IoT environment from UCD in 2008 and 2014 respectively. He was an Assistant Professor at Cyprus International University. His research interests include mobile computing, wireless sensor



publications, resulting in a h-index of 28. His specific research interests include the applicability of intelligent systems in context and opportunistic service provision, as well as participatory action research.

**MICHAEL J. O'GRADY** is a Senior Research Fellow at University College Dublin. He has over 30 years of experience both in industry and academia; his expertise has been internationally acknowledged by the ACM and IEEE through senior membership, as well as by international funding agencies as an expert reviewer. Dr O'Grady has published in a wide range of international fora, contributing to over 176 peer-reviewed



interests are in the areas of Artificial Intelligence and Multi-Agent Systems (MAS), Ubiquitous Computing and Wireless Sensor Networks. He is an established Principal Investigator with Science Foundation Ireland, having been one of the founders of the CLARITY centre (now INSIGHT) and the very successful CONSUS collaboration with Origin Enterprises in the area of Digital Agriculture.

**GREGORY M. P. O'HARE** is a Professor of Artificial Intelligence and Head of the School of Computer Science and Statistics at Trinity College Dublin (TCD) and a visiting Professor at University College Dublin (UCD). Prof. O'Hare has over 500 refereed publications of which over 100 are in high impact journals. He has edited 10 books and has a cumulative career research grant income of circa €82 million. His research

...