RoadNote: Automated Road Closure Detection using Urban Sensing

Raiful Hasan
Dept. of Computer Science
University of Alabama at Birmingham
Birmingham, AL 35294, USA
raiful@uab.edu

Ragib Hasan
Dept. of Computer Science
University of Alabama at Birmingham
Birmingham, AL 35294, USA
ragib@uab.edu

***Abstract*—Maps and navigation applications are essential tools in the modern era, especially for smartphone users. Navigation apps not only guide us on the correct path to the destination but also serve to find convenience and provide connectivity by sharing locations, travel status, and expert guidance. Map applications offer real-time updates which rely on crowdsourcing data from users, historical data, and advanced prediction algorithms. However, due to the dynamic nature of the urban environment, navigational apps fail to provide unscheduled road closure information. This study investigates erroneous situations and found 23 incidences where maps fail to navigate the closure information. We propose *ROADNote*, an automated system that accommodates urban sensors and provides closures update to users. *ROADNote* provides real-time traffic conditions by automated detections using future-generation commodity sensors. We built a prototype of *ROADNote*; after that, we conducted experiments to get real-time road-closer information by visual sensors (i.e., drone, camera). *ROADNote* facilitates to reduce of average travel time by 3.48 minutes and distance by more than 300 meters.**

***Index Terms*—road closure detection, smart city, automated detection, AV, UAV, smart mobility**

I. Introduction

Navigation apps have become an important part of our daily lives and offer several benefits. For example, it is a handy tool for traveling, searching, and real-time location sharing. According to a survey conducted in 2018, more than 77% of smartphone owners regularly use navigation apps, especially 87% of whom use maps when driving [1]. The monthly use information of maps application from multiple marketplaces shows the increasing growth of uses in present days [2]. In addition, services are changing from static apps to interactive real-time guides. Several navigation applications provide navigation services to users, including Google Maps[1], Waze[2], Apple Maps[3], MapQuest[4], and so on. Though the features are different in these applications, the primary functionality is identical; use the Internet connection to a GPS navigation system to provide instructions on arriving at a given destination. Some applications provide offline support but are unable to deliver real-time updates. However, now popular map applications (i.e., Google Maps, Apple Maps, and Waze) are almost accurate in estimating travel time.

[1]Google Maps - https://www.google.com/maps
[2]Waze - https://www.waze.com/live-map/
[3]Apple Maps - https://www.apple.com/maps/
[4]MapQuest - https://www.mapquest.com

Map applications show real-time traffic conditions by combining crowdsourced data from their user base or partners using roadside sensors and historical data. Then the prediction model applies to calculate the travel time, route, alternatives of suggested routes, etc. In addition, these applications use official data from local authorities and user feedback or reports to show road closures [3], [4]. Therefore, crowdsourced data from the community is crucial for Real Time Closure (RTC), where users manually report to the app. Different apps display the reported data to other users in accordance with their policies, and it varies on the platforms how long (e.g., a few minutes to months) it takes to authorize for others. For example, Waze collects data from its application users and reports it in a specific context; it depends entirely on the user community, where updates are quicker but may not be accurate.

On the other hand, cities are constantly bustling with activity. Residents frequently deal with traffic congestion, unscheduled road closures, road maintenance, and emergency management. In addition, sometimes festivals, games, and political rallies take place on the city street. The map applications often can not reliably accommodate these conditions. Maps that rely directly on user reports do not display these unannounced closures unless someone visits the location and manually reports to the application. In addition, the route closure is shown on the maps until people report it, even though the road resumed traffic way before that report. Therefore, some road closures remain unnoticed by maps, whereas some roads continuously show as closed. Though contemporary maps may provide an alternate route automatically, it takes longer to reach the destination. When there is one-way and single-lane traffic, the problem worsens. In addition, maps that rely upon local official data report to the maps and instantly re-route the traffic. However, it happen that the road might not close in reality. Therefore, crowdsourcing in dynamic urban contexts must be enhanced and automated using next-generation technologies and methodologies.

The rapid proliferation of smart commodity sensors has increased interest in developing applications that collect image and video data. In addition, the advancement in drones and the autonomous vehicles (AV) industry has created an opportunity to get real-time visual data in urban environments [5], [6]. Therefore, in addition to the primary task, drone fleets in aerial delivery and AV on the road could capture on-demand road conditions [7], [8]. In the future, when hundreds if not thousands of drones will operate flights in the sky could detect road closures on-demand and report to the maps. The same thing might happen for the AV. There are two possible scenarios: first, drones and vehicles might detect road closures and submit them automatically to maps; second, the map could request confirmation of any road closure reports for specified locations.

This study analyzes scheduled and unscheduled road closures in the urban environment. We conducted an observational study in 15 days time periods in a mid-sized US city. The study investigates how crowdsourcing data causes the discrepancy in road closure on maps. Then, we propose *ROADNote*, an automated system for reporting road closures driven by drones and autonomous vehicles. The *ROADNote* application notifies the users if there is any road closure on the route. We developed a proof of concep of the *ROADNote* using Android platform and conducted an experiment in a real road closure condition. Finally, we created a virtual traffic model using the Simulation of Urban Mobility (SUMO) to accommodate the described system [9].

Contributions: The contributions of this paper are as follows:

1) This study investigates instances where the maps application fails to give accurate data.
2) We assess 23 incidences and analyze the impact in terms of time and distance.
3) We proposed *ROADNote*, an automated road closure detection system based on an urban sensor. We implemented a proof of concept of *ROADNote* and assessed the feasibility.

Organization: The rest of the paper is organized as follows: in Section II, we discuss existing studies of road closures update. Section III presents the observational study details. In Section IV, we discuss the ROADNote architecture and the implementation details in Section V. Finally, Section VI presents the findings, and we conclude in Section VII.

II. Background

The quality of the navigational map is greatly influenced by delayed and missing detection of road closures. However, a handful of research has been conducted on map-inference/map-update, and existing solutions rely on crowdsourced data and user trajectories to identify road closures [10]. However, trajectory data often result in missed detection due to skewed data distribution and the co-occurrence of multiple errors with identical features [10], [11]. Some providers use static urban traffic sensors and local traffic data for detecting closures. However, the dynamic nature of the road networks and lack of mobility of sensors limit their uses. In addition, previous map-update research focuses more on detecting missing roads than identifying road closure events [12]. Some research classified road closures as a traffic anomaly and sought to identify them based on anomalous traffic situational changes [13]–[15]. These studies can be divided into two categories: threshold-based and statistics-based methods. Wang et al. [16] and Stanojevic et al. [17] developed threshold-based techniques for discovering higher and lower thresholds from historical data and identifying road closures by detecting abnormalities in traffic flow sequences in the time dimension. In contrast, by developing a Poisson process, Pietrobon et al. suggested a statistical-based technique for detecting road closures [11]. On the other hand, the traffic change depends on urban regions and roads, capturing the spatial and temporal dependences for traffic prediction is important. Therefore, some studies adopted graph-based structures for traffic prediction [18].

III. Road Closure in City

We conducted an observational study in a mid-sized US city where the total population in the metropolitan area is more than 1.1 million as of 2020 [19]. The study duration was 15 days, and we observed a total of 23 road closure incidents where there was a mismatch between real scenarios with map applications. All the experiment was held in or near the downtown area. For comparison, we chose three map applications (Google Maps, Waze, and Apple Maps) based on their market share. The selected region contains the urban university campus and the downtown of a metropolitan area. The estimated size of that area is $\approx$8 square miles; traffic varies on the locations, but one of the intersections has reported an average of 2048 (1480 + 548 in two cross street) vehicles per hour [20].

A. Methodology

A volunteer team roamed around this target area at different times of the day and took note (e.g., place, duration of the closure, photos, etc.) if there were any scheduled and unscheduled road closures. After spotting, we immediately compared the road's actual condition with map applications and only included them if there was an anomaly. In addition, we measured the time for re-routing to go to a point in map applications due to the road closure.

B. Closure Observation

Initially, we recorded 27 incidences of road closure information discrepancy. However, we discarded four cases where at least one map application has shown the closure information in a short period (within 30 minutes after we notice) if not real-time. We divided the rest of the 23 cases into three categories:

1) The Road Was Closed, Maps Failed To Show It: This was the most frequent occurrence we observed over 15 days (Figure 2(a)). There were 16 instances in which roads were closed for various reasons, yet map applications failed to provide this information and have never shown them. These closures lasted a minimum of 3 hours to a maximum of 18 hours.

2) The Road Was Open, Maps Showed Closed: There were four instances in which maps indicated that a road was closed, but in reality, they were open to traffic (Figure 2(b)). We assume that map applications may use local official

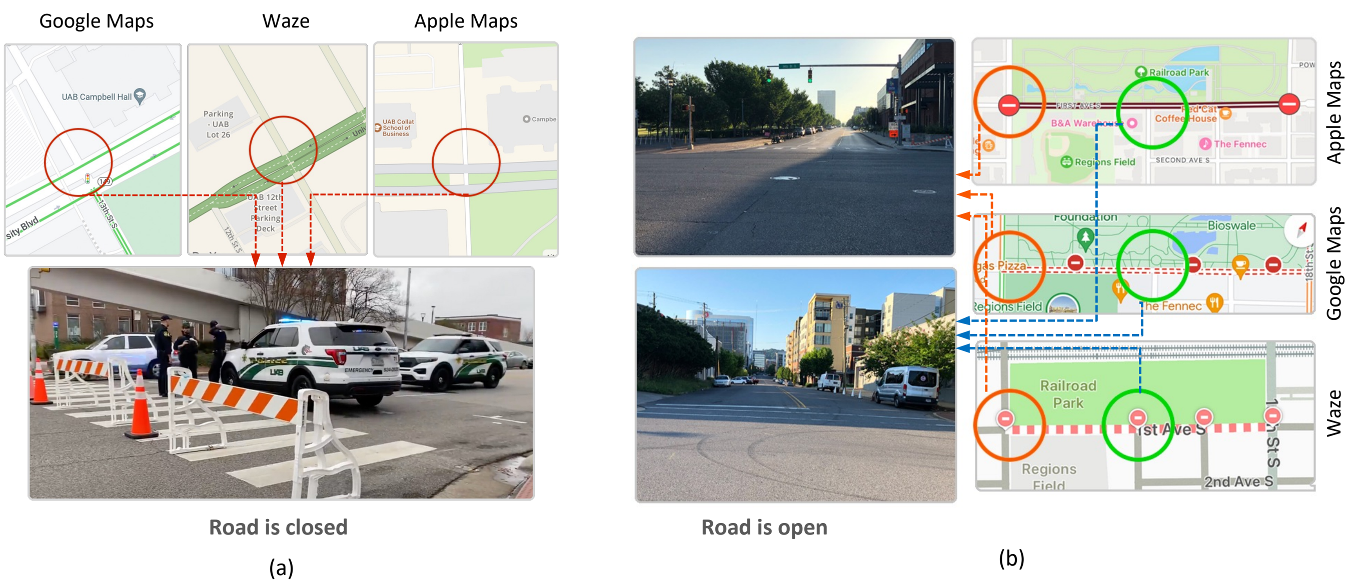

Fig. 1. Road closure observation. (a) shows that the road was closed for various reasons; however, the map applications never showed them. (b) shows that map applications indicate the road closure, but the road was open for traffic.

information for those closures. However, the actual road closure did not occur or lasted for a shorter duration than anticipated.

3) The Road Was Closed, Maps Updated After a Certain Time: We observed three incidences where the maps have been updated after a long period. For example, in one incidence, Waze shows a road closure after ≈20 hours from closing. Google and Apple Maps updated them within one hour after the update on Waze.

C. Impacts of Discrepancy

We investigate how the road closure information discrepancy impacts to traffic. The map applications reroute the travel path if there is any closure. As a result, rerouting often increases travel time. In addition, urban roadways experience mass traffic during peak hours; and rerouting incurs additional distance and expense along with travel time. Figure 2 illustrates how road closure discrepancy impacted traffic. Here, we present an instance where the roads around a stadium were blocked for ≈6 hours due to a sporting event. However, this closure information was not shown in the map applications. We experiment to identify the impact in terms of time and distance due to this discrepancy. We used Google Maps to get to a location that was initially 1.1 miles away by a vehicle. The maps initially indicated a route with an estimated travel time of four minutes. However, we were rerouted by map due to road closure and ended up with a distance of 1.9 miles and a travel time of 11 minutes. If maps had shown the closures in advance and avoided the primary route, we could have reached the destination in 1.2 miles and four minutes. This was a single-vehicle experiment where the average speed was 25 - 35 miles per hour; therefore, it may not represent the overall situation. The elapsed time and travel distance depend on routes, time on days, average vehicle speed, available alternatives, and traffic conditions. Therefore, we run a simulation model using SUMO to investigate the border impact on mass traffic. The results are discussed in Section VI.

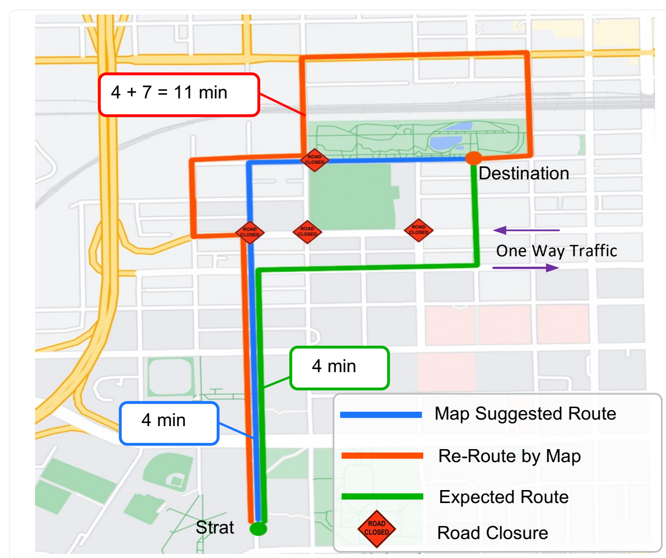

Fig. 2. Impacts on travel time and distance by road closure information discrepancy.

IV. ROADNOTE: AUTOMATED ROAD CLOSURE UPDATE

ROADNote is an automatic road closure monitoring system where the urban sensors notify and/or verify any occurrence. Current static and/or manual crowdsensing technologies need a direct line of sight in order to record any real-time closure. In addition, the static sensors cannot move to observe the object of interest directly. The proliferation of commodity drones and AVs may report these situations in real-time without human intervention. *ROADNote* consists of three components, and Figure 3 shows a high-level architecture.

A. On-Demand Urban Sensing

On-demand urban sensing combines static traffic and urban sensors, human crowdsourcing data, and dynamic sensors. Unmanned Aerial Vehicles (UAVs), especially drones and AVs, are the primary contributors to dynamic sensors. Though UAVs were initially introduced as military technology, nowadays, it proved potential in many sectors, including agriculture, traffic management, construction, etc [21], [22]. In addition, the delivery drone and air taxi are a reality now. Leading companies (e.g., Amazon, UPS, DHL, Walmart, etc.) have implemented drone delivery services. A study predicts that by 2026, more than a million drones will be delivering retail goods [23]. We are primarily interested in vision and computing capability, which can move on demand. The *ROADNote* technology offers these fleets a real-time road closure detection platform. UAVs and AVs may contribute in two ways; first, they can capture and transmit real-time information on road closures to the central server as they move. Second, they could verify the previously reported road closures. It is important to note that the automated detection and reporting will operate alongside the existing manual reporting. Therefore, *ROADNote* will provide an additional source of real-time data.

B. Processing Unit: Automated Detection Model

This is the extended version of the current central server. Existing one may analyze and verify (depending on the platform) manually submitted road closures using users' crowdsourced data. However, the extended version will process, detect, and verify the closures. The automated detection model is situated combinedly on the on-demand sensor and server-side. The detection model is responsible for identifying closures from image or video data. Real-time detection is computationally expensive, and commodity sensors may not run the complete analysis on the sensor side. Therefore, the sensors only detect the road closure signs and send the associated data (i.e., GPS locations, time, etc.) to the server. The server then runs the detection model extensively and identifies the road closures. In addition, the model verifies closures by cross-checking multiple reports and based on source rank (see IV-D). The verification can be conducted in two steps; first, the detection model decides the closures from the available incoming reports; second, the system sends a request for additional data to the nearest dynamic sensors, which may not be feasible with static traffic sensors.

C. Map Application

The map application is responsible for the user interface and offers real-time navigation based on the user's location. The map application only shows the closures after the detection model thoroughly verifies them. However, there is an option to report an incident or any emergency in cities manually. In addition, if the user gives consent, *ROADNote* tracks the position and provides a context-based warning if it detects any closure on the route, even if the user does not use the applications.

D. Source Rank and Verification Process

ROADNote must verify crowdsourcing data to assure the accuracy of data received from several sources. *ROADNote* accommodates two types of reporters, human users and dynamic sensors (e.g., UAV, AV, etc.). We defined both these types as users and adopted a user rating verification system based on the previous history. We classified users into three categories based on their ratings: (i) Super User: the most trustworthy user group whose data correctness was confirmed to be accurate. *ROADNote* approves and shows the reported content without modification and further verification in real-time. The dedicated UAVs and AVs are also super users. (ii) Associate User: this group of users is not dedicated to the maps but provides reliable data. Therefore, *ROADNote* defines the high-rated human user and commercial UAVs and AVs as the associate user. After getting data from these users, *ROADNote* shows on the map service if at least one super user approves it. (iii) Regular User: *ROADNote* publishes closure data of the regular user after getting approval by a threshold number of the super user (at least 1) or associate user (at least 3).

V. Implementation

We have implemented a proof of concept for ROADNote by adopting a semi-automated process and utilizing different techniques to implement various components of ROADNote. Therefore, the implementation procedure is divided into three phases. First, the objective is to identify road closures from image data captured by UAVs and AVs (automated reporters). The advancements in computer vision and the AV industry have facilitated the detection of objects in real-time. For automated road closure detection, we used the YOLOv5 [24] object detection algorithm on the custom dataset. We manually captured images and videos of road closures for training the model. We used DJI Mini 2 and GoPro Hero9 Black for capturing the image and video. All the images we used here were collected in the city of Birmingham, Alabama, USA. After prepossessing (i.e., annotation, class define, etc.) and training the model, we successfully detected the road closure with confidence greater than 50 (Figure 4). Despite the short sample size, the model verified that we could detect closures from AV-captured images in real-time.

Second, we used Amazon Web Services (AWS) to implement the server-side functionality. After detecting the road closure, the automatic system notifies the central server. The server stores image, time, GPS location, and status in the database. Initially, every report status showed unconfirmed but marked as confirmed after getting verification. The server sends a notification to users (super user, associate user, and regular user) to verify the status of closures. Finally, we have implemented a mobile application for the Android platform for map services called *ROADNote*. We use OpenStreetMap, and its tool suite for the base map with related read/write and database import/export activities. *ROADNote* faces the road closures information

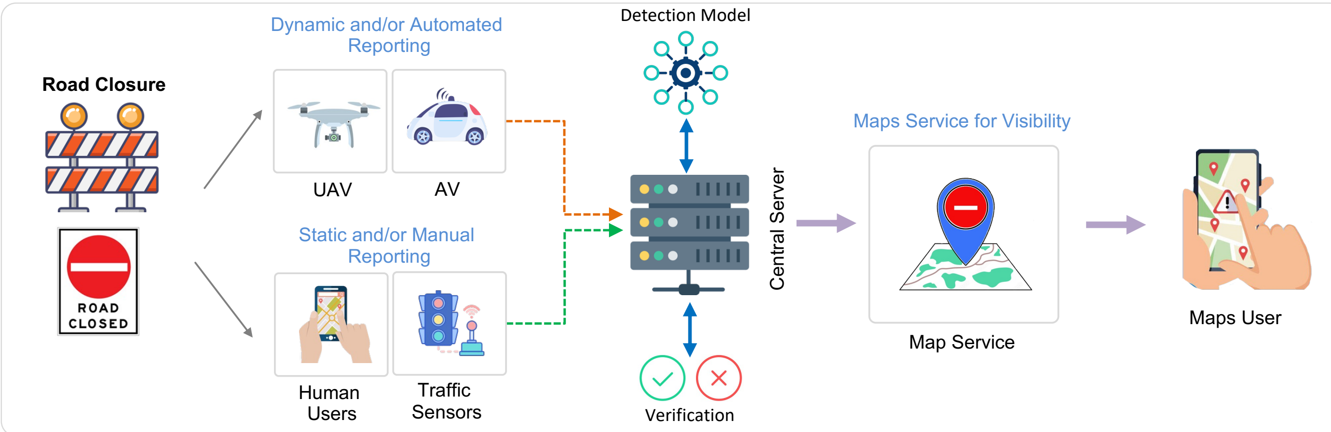

Fig. 3. High Level architecture of *ROADNote*. The "Automated/Manual" reporting denotes the reporting of closures by machines and humans.

Fig. 4. Detection of road closures using YOLOv5 on collected photos.

from the server and adds them to the map with the tag. We implemented a background service to perform the information-facing operations and populate them in maps using a foreground thread.

VI. Findings

Our study on road closure discrepancy in a mid-size US city for 15 days identified 23 incidences which we classified into three subcategories. We tracked the duration of these incidents after we noticed them. In 47.83% of instances, roads were closed for between three and nine hours; roads were closed for 10 to 14 hours in 30.43% of the times (Figure 5 (a)). Most of these closures happened during the daytime and were caused by several events.

We experimented on roadways of closure impacts during the maps discrepancy observation (Figure 2). However, it is challenging to accommodate the variable traffic factors manually (i.e., speed of vehicles, direction, traffic congestion, route selection, etc.). Therefore, we implemented two traffic roadways models using SUMO, identical to the original street and signals. SUMO has a feature to control the traffic externally and allow real-time input to the system. We utilize Traffic Control Interface (TraCI) and Python interface, which facilitates the communication between model and server. We implemented models with four variable control factors: direction, speed, distance, and congestion. Figure 5 (b) shows the elapsed time of 50 vehicles where the system randomly selected all four above controls. The suggested route is initially provided by maps when the maps do not have the closure information. The average suggested time to reach a destination is 6.15 minutes, where the standard deviation (SD) is 1.51. However, the maps rerouted vehicles due to closures, increasing the total elapsed time. As a result, the total elapsed time increased to 9.63 minutes (SD is 2.02). Therefore, each vehicle spends an additional 3.48 minutes on average due to a single unreported road closure at a particular location. In comparison, *ROADNote* offers an alternate route with an average elapsed duration of 6.21 minutes (SD = 1.9) after the models get data from the server.

Figure 5 (c) presents the comparison of the average distance to the destination due to the road closures. The average suggested distance is 1302.35 meters before rerouting. After reroute, the average distance increased to 1965 meters. Therefore, vehicles traveled an additional 662 meters on average due to unreported road closures. In contrast, the ROADNote suggested route shows 1639 meters on average to reach the destination. Finally, Figure 5 (d) depicts the cumulative effects of unreported closures endured within a specified period. We calculated the cumulative time and distance consequences of a five-hour road closure. For example, if the maps fail to show road closure for one hour, the impacted vehicles will spend additional 3.24 hours and would travel additional 18.29 miles. If it sustains for three and five hours, the total vehicle's additional time will increase to 18.43 hours and 58.26, respectively; the additional distance will be 95.64 miles and 136.15 miles.

VII. Conclusion and Future Work

This study presents the observational data on road closure discrepancy. Due to the city's dynamic nature, a variety of unanticipated events/situations result in road closures. As modern map applications rely on crowdsourced data from users to detect closures, closures can get undetected in map services. We present the study data of 23 incidences where the map applications fail to show. In addition, this study proposes

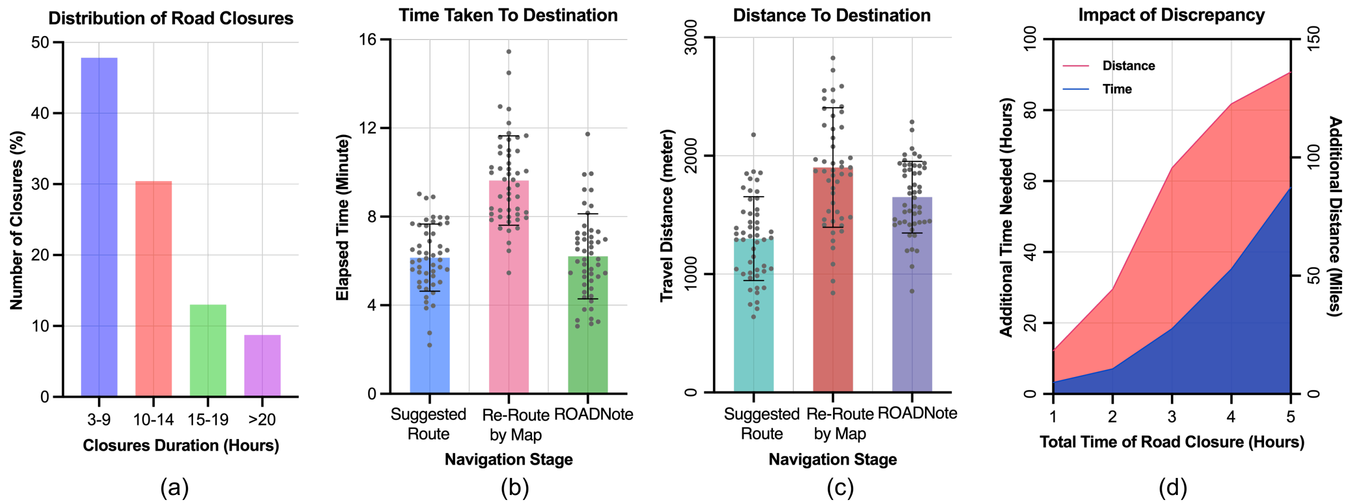

Fig. 5. Findings of road closures observation and *ROADNote*; (a) Shows the closure incidence time interval distribution; (b) presents the elapsed time of maps application and *ROADNote*; (c) shows the distnace comparison of the suggested route, re-routed by maps, and *ROADNote*; (d) illustrates the impact of road closure discrepancy in terms of time and distance if the situation sustains up to 5 hours.

ROADNote, an automated road closures detection system using AV and UAV in addition to manual crowdsourcing. Experiments found that by ROADNote, vehicles could save on average 3.48 minutes of travel times and can travel 326 meters less. In the future, we will investigate universal road closure in addition to sign-based road closure. Next steps will also be taken towards automating the road closure directly from UAV fleets without server-based computation assistance.

References

[1] R. Panko, "The Popularity of Google Maps: Trends in Navigation Apps in 2018." https://themanifest.com/app-development/trends-navigation-apps, 2018. [Online; Last Accessed June 18, 2022].

[2] A. He, "People Continue to Rely on Maps and Navigational Apps." https://www.emarketer.com/content/people-continue-to-rely-on-maps-and-navigational-apps-emarketer-forecasts-show, 2019. [Online; Last Accessed June 18, 2022].

[3] J. Lau, "Google Maps 101: How AI Helps Predict Traffic and Determine Routes." https://www.emarketer.com/content/people-continue-to-rely-on-maps-and-navigational-apps-emarketer-forecasts-show, 2020. [Online; Last Accessed June 20, 2022].

[4] J. J. Rojas, "Using Traffic Data with Maps and Routes." https://developer.tomtom.com/blog/build-different/using-traffic-data-maps-and-routes, 2021.

[5] A. Gohari, A. B. Ahmad, R. B. A. Rahim, A. Supa'at, S. Abd Razak, and M. S. M. Gismalla, "Involvement of Surveillance Drones in Smart Cities: A Systematic Review," *IEEE Access*, 2022.

[6] R. Fan, M. J. Bocus, Y. Zhu, J. Jiao, L. Wang, F. Ma, S. Cheng, and M. Liu, "Road Crack Detection using Deep Convolutional Neural Network and Adaptive Thresholding," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 474–479, IEEE, 2019.

[7] C. Kyrkou and T. Theocharides, "EmergencyNet: Efficient Aerial Image Classification for Drone-based Emergency Monitoring Using Atrous Convolutional Feature Fusion," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 1687–1699, 2020.

[8] M. A. Hoque, R. Hasan, and R. Hasan, "R-CAV: On-Demand Edge Computing Platform for Connected Autonomous Vehicles," in *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, pp. 65–70, IEEE, 2021.

[9] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*, ThinkMind, 2011.

[10] S. Cai, T. Wu, J. Mao, and C. Jin, "Road closure detection based upon multi-feature fusion," in *Proceedings of the 29th International Conference on Advances in Geographic Information Systems*, pp. 354–364, 2021.

[11] D. Pietrobon, A. P. Lewis, and G. S. Heverly-Coulson, "An algorithm for road closure detection from vehicle probe data," *ACM Transactions on Spatial Algorithms and Systems (TSAS)*, vol. 5, no. 2, pp. 1–13, 2019.

[12] P. Chao, W. Hua, and X. Zhou, "Trajectories know where map is wrong: an iterative framework for map-trajectory co-optimisation," *World Wide Web*, vol. 23, no. 1, pp. 47–73, 2020.

[13] S. Chawla, Y. Zheng, and J. Hu, "Inferring the root cause in road traffic anomalies," in *2012 IEEE 12th International Conference on Data Mining*, pp. 141–150, IEEE, 2012.

[14] H. Zhang, Y. Zheng, and Y. Yu, "Detecting urban anomalies using multiple spatio-temporal data sources," *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 2, no. 1, pp. 1–18, 2018.

[15] L. X. Pang, S. Chawla, W. Liu, and Y. Zheng, "On detection of emerging anomalous traffic patterns using gps data," *Data & Knowledge Engineering*, vol. 87, pp. 357–373, 2013.

[16] Y. Wang, X. Liu, H. Wei, G. Forman, C. Chen, and Y. Zhu, "Crowdatlas: Self-updating maps for cloud and personal use," in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pp. 27–40, 2013.

[17] R. Stanojevic, S. Abbar, S. Thirumuruganathan, G. D. Francisci Morales, S. Chawla, F. Filali, and A. Aleimat, "Road network fusion for incremental map updates," in *LBS 2018: 14th International Conference on Location Based Services*, pp. 91–109, Springer, 2018.

[18] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2019.

[19] Wikipedia, "Birmingham, Alabama, USA." https://bit.ly/3NQfx1t, 2022.

[20] R. Hasan, M. A. Hoque, Y. Karim, R. Griffin, D. Schwebel, and R. Hasan, "Smartphone-based Distracted Pedestrian Localization using Bluetooth Low Energy Beacons," in *2020 SoutheastCon*, pp. 1–2, IEEE, 2020.

[21] P. Radoglou-Grammatikis, P. Sarigiannidis, T. Lagkas, and I. Moscholios, "A compilation of uav applications for precision agriculture," *Computer Networks*, vol. 172, p. 107148, 2020.

[22] S. Srivastava, S. Narayan, and S. Mittal, "A survey of deep learning techniques for vehicle detection from uav images," *Journal of Systems Architecture*, vol. 117, p. 102152, 2021.

[23] L. Goasduff, "Why Flying Drones Could Disrupt Mobility and Transportation Beyond COVID-19." https://www.gartner.com/smarterwithgartner/why-flying-drones-could-disrupt-mobility-and-transportation-beyond-covid-19, 2020.

[24] G. Jocher, "YOLOv5 by Ultralytics." https://github.com/ultralytics/yolov5, 2020.