2020

Addressing Rogue Vehicles by Integrating Computer Vision, Activity Monitoring, and Contextual Information

Brook Abegaz

David Chan-Tin

Neil Klingensmith

George K. Thiruvathukal

Addressing Rogue Vehicles by Integrating Computer Vision, Activity Monitoring, and Contextual Information

Brook Abegaz
Loyola University Chicago

Eric Chan-Tin
Loyola University Chicago

Neil Klingensmith
Loyola University Chicago

George K. Thiruvathukal
Loyola University Chicago

ABSTRACT

In this paper, we address the detection of rogue autonomous vehicles using an integrated approach involving computer vision, activity monitoring and contextual information. The proposed approach can be used to detect rogue autonomous vehicles using sensors installed on observer vehicles that are used to monitor and identify the behavior of other autonomous vehicles operating on the road. The safe braking distance and the safe following time are computed to identify if an autonomous vehicle is behaving properly. Our preliminary results show that there is a wide variation in both the safe following time and the safe braking distance recorded using three autonomous vehicles in a test-bed. These initial results show significant progress for the future efforts to coordinate the operation of autonomous, semi-autonomous and non-autonomous vehicles.

CCS CONCEPTS

• **Security and privacy → Distributed systems security**.

KEYWORDS

Autonomous Vehicles; Rogue Vehicle; Monitoring; Sensors

ACM Reference Format:
Brook Abegaz, Eric Chan-Tin, Neil Klingensmith, and George K. Thiruvathukal. 2020. Addressing Rogue Vehicles by Integrating Computer Vision, Activity Monitoring, and Contextual Information. In *12th International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI '20 Adjunct), September 21–22, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3409251.3411724

1 INTRODUCTION

There has been significant progress to develop and enhance autonomous vehicles, especially in control systems and computer vision, with the ultimate goal being to enable the vehicles to self-drive [2–5]. This progress has allowed for many autonomous and semi-autonomous vehicles at various levels of automation to be operating on the roads today. More recently, however, there has been some work to address security of autonomous vehicles, especially on the computer vision aspect [1]. This includes adversarial training and including unexpected objects in plain view of the vehicle's camera to modify its object recognition.

Addressing the security of autonomous vehicles is critical as they become more widespread. A *rogue vehicle* is one which is not behaving correctly — this could be due to faulty or compromised sensors—for example, following too closely with the vehicle in front, whether it is autonomous or not. In densely populated cities such as New York City, Los Angeles, Chicago, etc., we often observe ordinary manually-operated vehicles driving erratically: swerving in traffic, driving much faster or much slower than other vehicles, rolling through stop signs, etc. As traffic densities continue to rise, we can expect that unsafe driving patterns will become more prevalent, threatening the safety of people in the vehicle, in other vehicles, and pedestrians on the road. Introducing autonomous vehicles into the traffic patterns will create more unknowns, as human drivers must learn to interact with autonomous systems on the same roads.

Instead of securing individual components or each communication protocol of an autonomous and semi-autonomous vehicular system, we consider a different perspective of addressing vehicular security. Securing each individual component is not scalable as new components by different manufacturers get introduced. A vehicle, whether autonomous, semi-autonomous, or non-autonomous, should only exhibit certain behaviors: e.g., drive within certain speed ranges, switch lanes when necessary, make safe turns, etc. The holistic approach is to have every vehicle or a significant subset of vehicles *monitor* other vehicles and categorize their behavior, building on the tradition of distributed systems research by establishing *consensus*. By using voting mechanisms, the monitoring vehicles can make a determination of if a vehicle is rogue and conduct an election to make a determination of how to respond to the rogue vehicle. Invalid or malicious behavior can then be reported. A detected bad behavior could be due to one hacked component (e.g. hacked accelerometer) or one faulty component (e.g. smudge on camera). Assuming that the majority of autonomous vehicles are behaving correctly and correctly functioning cameras and sensors, this approach can achieve a higher scale of security and be used to take an appropriate action.

A major challenge in this line of work is that—fortunately for passengers and drivers—rogue autonomous vehicles are not yet widely prevalent on the roads. This lack of exemplar behavior makes the task of distinguishing rogue vehicles difficult. We do, however, have the occasional opportunity to observe rogue behavior from human-controlled vehicles on the road: non-autonomous vehicles

AutomotiveUI '20 Adjunct, September 21–22, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-8066-9/20/09.
https://doi.org/10.1145/3409251.3411724

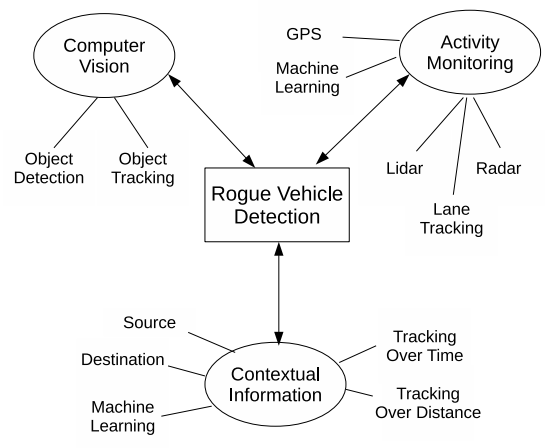

Figure 1: Our design with the three main components along with their requirements.

or bicycles rolling through stop signs and red lights, vehicles weaving through traffic, etc. All of these examples provide a glimpse of the possible behavior that rogue autonomous vehicles could exhibit.

The overall goals and contributions of this research are

- Introduce a new approach to secure autonomous vehicles by detecting rogue vehicle behavioral patterns.
- Introduce a realistic threat model for autonomous vehicle and rogue vehicles.

Figure 1 shows an overview of our design.

2 THREAT MODEL

In this work, our goal is to identify software on autonomous vehicles that is not functioning properly, either because of (a) the software is buggy, (b) it is intentionally compromised by a malicious agent, or (c) the vehicle is experiencing hardware failure(s) that cause software malfunctions. Our goal is to identify this broad class of malfunctions in real time by observing the behavior of each vehicle on the road. In our threat model, there are two classes of rogue vehicles: those that are intentionally violating traffic rules and those that are experiencing undetected malfunctions which cause them to unintentionally violate traffic rules. In either case, we define a rogue vehicle as one that is experiencing a malfunction that causes it to behave in an unexpected fashion.

We confine our definition of rogue vehicles to those that are experiencing software or hardware failures that cause malfunctions under normal road conditions. For example, we assume that an adversary is not attempting to compromise vehicles by defacing or damaging environmental controls like stop signs, stop lights, etc. We also assume that the machine learning process works as expected and there are no adversaries or tampering during the learning process. The machine learning algorithm is used to determine normal behavior and predict that a vehicle is rogue since it is expecting abnormal behavior.

The rogue vehicles could have one or more faulty components, for example, a broken LIDAR sensor or a smudge on the camera. Alternatively, the rogue vehicles could also have been hacked. In our threat model, we are not considering how they are hacked or faulty, but consider alterations from the general behavior of the vehicle.

All vehicles on the road, whether autonomous or manually-operated, can act as rogue vehicle detectors. Our assumption is that the majority of autonomous vehicles would behave correctly, which could be determined by establishing consensus. We also assume that vehicles have the necessary hardware to sense their environment and that they have reasonable wireless communications capabilities (a reasonable assumption in urban areas with 5G, where autonomous vehicles are likely to be deployed).

Minimum Distance before Collision			
	Route #1	Route #2	Route #3
AV2 -> AV1	0.3048m	0.254m	0.127m
AV3 -> AV1	0.1905m	0.1778m	0.2159m
Exposure Time during Travel			
	Route #1	Route #2	Route #3
AV2 -> AV1	2.13 seconds	3.33 seconds	2.4 seconds
AV3 -> AV1	2.66 seconds	2.40 seconds	2.25 seconds

Table 1: Measurements of Braking Distance Scenarios.

3 EXPERIMENTAL RESULTS

We performed an experiment involving three prototype autonomous vehicles. One autonomous vehicle was following another autonomous vehicle. The first autonomous vehicle braked to a complete stop, during which time the safe braking distance for the autonomous vehicle that was following it was recorded. We repeated this experiment three times with two different following autonomous vehicles. Table 1 shows the distance needed for the following autonomous vehicle to brake to a complete stop without colliding with the car in front of it. The time needed to brake that minimum distance is also recorded. The table shows that there is a variation in both the minimum distance before collision and the time to brake. This implies that any rogue vehicle detection needs to take into account these deviations and possible errors during monitoring.

Our experimental results show that detecting rogue vehicles is a non-trivial problem. The sensors in the three types of vehicles have different ranges of accuracy and sensitivity. The fact that such differences require coordination among autonomous, semi-autonomous and non-autonomous vehicles presents a significant distributed data-management challenge.

4 DISCUSSION AND FUTURE WORK

We have laid out our vision for the future of secure operation of autonomous vehicles with a focus on detecting rogue vehicles in operation. This is a low-cost solution, leveraging the information from nearby autonomous vehicles. To achieve this fully, a more realistic test-bed needs to be created with autonomous vehicles equipped with cameras, sensors, and short-range communications capabilities.

Each of our components – computer vision, activity monitoring, and contextual information – would then need to be slowly

incorporated and tested for efficiency and effectiveness in the test-bed. Autonomous vehicles exhibiting one or more–and possibly different–rogue behaviors will need to be identified. False-positive errors are inherent in any kind of detection scheme and will need to be taken into account in the larger test-bed. We also plan to utilize machine learning so appropriate training data will need to be collected. Training data itself presents a non-trivial challenge as visual data in particular requires substantial storage that must be maintained for repeated analysis as vision algorithms are improved over time.

This work can also be extended to detect rogue objects, such as rogue bicycles, rogue pedestrians, even rogue unmanned aerial vehicles or drones. This is feasible as connected autonomous drones and autonomous delivery robots are being deployed. Moreover, pedestrians and bicyclists could have an always-on Internet connection. Therefore, smartphones could be used to capture data to detect rogue pedestrians and bicycles. While these always-oh Internet connections are mostly delivered today via smartphones, with one connected mobile device per human, there is every reason to expect that more and more always-on wireless devices will be connected using 4G/5G connections, which would allow rogue pedestrians and bicycles to participate using whatever device (and on board) happens to be available.

The accelerometer and the GPS units in the smart phone could be used to verify the secure operation of the autonomous vehicle. This could be done by comparing the accelerometer and the GPS readings on the mobile phone with those readings recorded by sensors in the autonomous vehicle. This could give an additional way of checking if the vehicle's sensors are functioning properly and if they are sending accurate signals or if they have been compromised due to a faulty component or operation. Accordingly, the security verification system will not have to rely entirely on expensive sensor equipment attached to the vehicle that may not be detached. Instead, a low-cost method of using the built-in sensors of smart phones could provide a reliable alternative solution to the security problem of addressing rogue autonomous vehicles.

REFERENCES

[1] Genevieve Bell and Paul Dourish. 2007. Yesterday's tomorrows: notes on ubiquitous computing's dominant vision. *Personal and ubiquitous computing* 11, 2 (2007), 133–143.

[2] Shubham Jain, Carlo Borgiattino, Yanzhi Ren, Marco Gruteser, Yingying Chen, and Carla Fabiana Chiasserini. 2015. LookUp: Enabling Pedestrian Safety Services via Shoe Sensing. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (Florence, Italy) *(MobiSys '15)*. Association for Computing Machinery, New York, NY, USA, 257–271. https://doi.org/10.1145/2742647.2742669

[3] Sugang Li, Xiaoran Fan, Yanyong Zhang, Wade Trappe, Janne Lindqvist, and Richard E. Howard. 2017. Auto++: Detecting Cars Using Embedded Microphones in Real-Time. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article Article 70 (Sept. 2017), 20 pages. https://doi.org/10.1145/3130938

[4] C. Lin, Y. Chen, J. Chen, W. Shih, and W. Chen. 2016. pSafety: A Collision Prevention System for Pedestrians Using Smartphone. In *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*. IEEE, USA, 1–5. https://doi.org/10.1109/VTCFall.2016.7881183

[5] Luyang Liu, Cagdas Karatas, Hongyu Li, Sheng Tan, Marco Gruteser, Jie Yang, Yingying Chen, and Richard P. Martin. 2015. Toward Detection of Unsafe Driving with Wearables. In *Proceedings of the 2015 Workshop on Wearable Systems and Applications* (Florence, Italy) *(WearSys '15)*. Association for Computing Machinery, New York, NY, USA, 27–32. https://doi.org/10.1145/2753509.2753518